

Vysoká škola logistiky o.p.s.

**Informační podpora prodejního kanálu
malé firmy**

(Bakalářská práce)



Vysoká škola
logistiky
o.p.s.

Zadání bakalářské práce

student

Maxmilian Fišárek, DiS.

studijní program
specializace

Logistika
Informatika pro logistiku

Vedoucí Katedry bakalářského studia Vám ve smyslu čl. 22 Studijního a zkušebního řádu Vysoké školy logistiky o.p.s. pro studium v bakalářském studijním programu určuje tuto bakalářskou práci:

Název tématu: **Informační podpora prodejního kanálu malé firmy**

Cíl práce:

Na základě analýzy možných distribučních kanálů krejčovské dílny navrhnete vhodnou informační podporu. Porovnat jednotlivé způsoby řešení a navržené řešení zhodnotit.

Zásady pro vypracování:

Využijte teoretických východisek oboru logistika. Čerpejte z literatury doporučené vedoucím práce a při zpracování práce postupujte v souladu s pokyny VŠLG a doporučeními vedoucího práce. Části práce využívající neveřejné informace uveďte v samostatné příloze.

Bakalářskou práci zpracujte v těchto bodech:

Úvod

1. Distribuční procesy malé firmy
2. Informační podpora malé firmy
3. E-shop a jeho realizace
4. Návrh řešení
5. Zhodnocení

Závěr

Rozsah práce: 35 – 50 normostran textu

Seznam odborné literatury:

Gros, I., Barančík, I., Čujan, Z.: Velká kniha logistiky. Praha: VŠCHT, 2016. ISBN 978-80-7080-952-5.

Macurová, P. Klabusajová, N. Tvrdoň, L.: Logistika, 2. upravené a doplněné vydání, SOET, vol. 16. Ostrava: VŠB – TU, Ostrava 2018. ISBN 978-80-248-4158-8.

Mařík, V. et.al. Národní iniciativa Průmysl 4.0. Konfederace zaměstnavatelských a podnikatelských svazů 2016 [online]. [cit. 2016-10-01] Dostupné z: <http://kzps.cz/wp-content/uploads/2016/02/kzps-cr.pdf>.

Vedoucí bakalářské práce:

doc. Dr. Ing. Oldřich Kodým

Datum zadání bakalářské práce:

31. 10. 2020

Datum odevzdání bakalářské práce:

6. 5. 2021

Přerov 31. 10. 2020



Ing. et Ing. Iveta Dočkalíková, Ph.D.
vedoucí katedry



prof. Ing. Václav Cempírek, Ph.D.
rektor

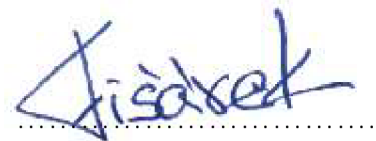
Čestné prohlášení

Prohlašuji, že předložená bakalářská práce je původní a že jsem ji vypracoval samostatně. Prohlašuji, že citace použitých pramenů je úplná a že jsem v práci neporušil autorská práva ve smyslu zákona č. 121/2000 Sb., o autorském právu, o právech souvisejících s právem autorským a o změně některých zákonů (autorský zákon), ve znění pozdějších předpisů.

Prohlašuji, že jsem byl také seznámen s tím, že se na mou bakalářskou práci plně vztahuje zákon č. 121/2000 Sb., o právu autorském, právech souvisejících s právem autorským a o změně některých zákonů (autorský zákon), ve znění pozdějších předpisů, zejména § 60 – školní dílo. Beru na vědomí, že Vysoká škola logistiky o.p.s. nezasahuje do mých autorských práv užitím mé bakalářské práce pro pedagogické, vědecké a prezentační účely školy. Užiji-li svou bakalářskou práci nebo poskytnu-li licenci k jejímu využití, jsem si vědom povinnosti informovat o této skutečnosti Vysokou školu logistiky o.p.s.

Prohlašuji, že jsem byl poučen o tom, že bakalářská práce je veřejná ve smyslu zákona č. 111/1998 Sb., o vysokých školách a o změně a doplnění dalších zákonů (zákon o vysokých školách), ve znění pozdějších předpisů, zejména § 47b. Taktéž dávám souhlas Vysoké škole logistiky o.p.s. ke zpřístupnění mnou zpracované bakalářské práce v její tištěné i elektronické verzi. Tímto prohlášením souhlasím s případným použitím této práce Vysokou školou logistiky o.p.s. pro pedagogické, vědecké a prezentační účely.

V Přerově, dne 18. 01. 2021



Jiří Šávek

Poděkování

Tímto bych chtěl poděkovat několika lidem. Prvně chci poděkovat doc. Dr. Ing. Oldřichovi Kodymovi za možnost tuto práci zpracovat a za podporu a pobízení v práci na ní. Dále bych chtěl poděkovat své matce za podporu při studiu a čas mně věnovaný při psaní této práce, obzvláště při zpracovávání analýzy její obchodní značky. Dále bych chtěl poděkovat Mgr. Evě Petrové, Bc. Kamilovi Stuchlíkovi a Mgr. Julii Vondrákové za podporu a inspiraci ke studiu.

Anotace

Tato práce pojednává o vypracování marketingové SWOT analýzy malé firmy a následné zpracování výsledných doporučení pro posílení pozice značky na trhu. Nastiňuje problematiku distribuce, v souvislosti s požadavky vlastníka, probírá téma marketingu v online prostředí. Analyzuje možné způsoby pro vývoj softwaru, konkrétně aplikace e-shopu. Nakonec popisuje způsob řešení tohoto nového distribučního kanálu.

Klíčová slova

Distribuce, SWOT analýza, Digitální marketing, Open Source, E-shop s výrobou na poptávku

Annotation

This work deals with the elaboration of a marketing SWOT analysis of a small company and the subsequent elaboration of the resulting recommendations for strengthening the brand's position on the market. It outlines the issue of distribution, in connection with the requirements of the owner, discusses the topic of marketing in the online environment. It analyzes possible ways for software development, specifically e-shop application. Finally, it describes how to solve this new distribution channel.

Keywords

Distribution, SWOT analysis, Digital marketing, Open source, E-shop on demand

Obsah

Úvod.....	10
1 Distribuční procesy malé firmy	11
1.1 Více stupňový distribuční systém	11
1.1.1 Maloobchod	12
1.1.2 Velkoobchod	12
1.1.3 Distributor	12
1.1.4 CrossDock.....	13
1.2 Přímý distribuční systém.....	13
1.3 Volba rozšíření distribučních kanálů firmy.....	13
1.3.1 Charakteristika značky	14
1.3.2 Historie značky	14
1.3.3 SWOT analýza a její charakteristika	14
1.3.4 Historie SWOT analýzy	15
1.3.5 Použití SWOT analýzy	15
1.3.6 Členění SWOT analýzy	16
1.3.7 Výstupy SWOT analýzy	17
1.3.8 Vypracování SWOT analýzy	17
1.3.9 Výsledek SWOT analýzy.....	18
2 Informační podpora malé firmy.....	19
2.1 Digitální marketing	19
2.1.1 Historie digitálního marketingu	19
2.1.2 Význam digitálního marketingu	20
2.1.3 Výhody digitálního marketingu	20
2.1.4 Sociální média.....	21
2.2 SEO – Search Engine Optimization.....	22
2.2.1 Etika v SEO	23

2.2.2	SEM a rozdíly od SEO.....	24
2.3	Východiska pro značku.....	24
3	E-shop a jeho realizace.....	25
3.1	Požadavky na konkrétní řešení.....	25
3.2	Jazyk C#.....	25
3.2.1	.NET framework.....	27
3.3	MVC architektura.....	28
3.4	Open Source versus vlastní zdrojový kód.....	29
3.4.1	Open Source.....	29
3.4.2	Výhody vlastního kódu.....	30
3.5	Výstup analýzy prostředků.....	30
4	Návrh řešení.....	32
4.1	Postup při návrhu konceptu aplikace.....	32
4.2	IDE – vývojové prostředí.....	33
4.2.1	Nastavení a instalace Visual Studia 2019.....	34
4.2.2	Založení nového projektu.....	34
4.3	Databáze.....	34
4.3.1	Structured Query Language.....	34
4.3.2	Entity Framework.....	35
4.3.3	Linq.....	36
4.3.4	Vytvoření a úprava databáze.....	36
4.4	Správa uživatelských účtů.....	37
4.5	Řešení kapacity dílny.....	37
4.5.1	Datum expedice.....	38
4.6	Řešení rozdílné velikosti produktů.....	38
4.7	Detail produktu.....	39
4.8	Správa objednávek.....	39

4.9	Generování faktury	40
4.10	Zasílání emailových zpráv	41
4.11	Platební brána Paypal	42
4.11.1	Další platební metody	42
5	Zhodnocení	43
	Závěr	44
	Seznam zdrojů.....	45
	Seznam grafických objektů.....	47
	Seznam zkratk	48
	Seznam příloh	50

Úvod

Cílem této práce je analýza přípustných distribučních kanálů oděvní značky a realizace zvoleného nového distribučního kanálu a zároveň analýza možné informační podpory prodeje.

V teoretické části této práce je nastíněna problematika distribučních procesů malé firmy, konkrétně krejčovské dílny designové oděvní značky. Při volbě vytváření nové distribuční cesty je třeba zohlednit mnoho aspektů, jako je exkluzivita značky, jedinečnost designu a kvalita ručního zpracování jednotlivých výrobků.

Dále je nastíněna problematika propagace v online prostředí pomocí různých prostředků digitálního marketingu. Je zmíněna i problematika optimalizace nalezitelnosti a její základní techniky.

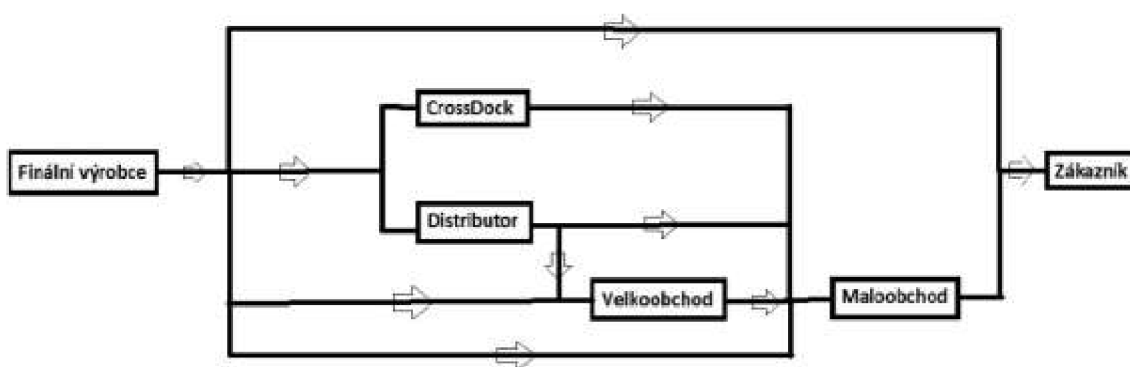
Na základě marketingové analýzy je pak zvolena možnost rozšíření distribučních cest vyhovující vlastníkově této značky. Jedná se o zřízení vlastního e-shopu plně pod správou vlastníka značky a díky tomu i plné kontrole celého nového distribučního kanálu.

V části praktické je pak řešeno možné zpracování libovolného softwaru za určitým záměrem. Jsou nastíněny možnosti celého naprogramování i použití a úprava open-source řešení a jednotlivé výhody a úskalí těchto způsobů řešení.

Konečně je řešeno konkrétní provedení zhotovení e-shopu pro zvolený účel. Včetně ojedinělého požadavku výroby na objednávku místo správy kusů zboží na skladě a řešení výrobní kapacity dílny v rámci počítání dodacích lhůt. Probírány jsou především požadavky, kterými se aplikace e-shopu pro značku MAXFRED liší od vzorové aplikace poskytované webem ABC tutoriál a prvky, které ve vzorové aplikaci řešené nejsou a jsou pro chod e-shopu nezbytné.

1 Distribuční procesy malé firmy

Distribuční proces libovolné firmy může nabývat několika forem. Jedním z hledisek struktury distribučního procesu je počet stupňů, které distribuční systém dané společnosti má. V základu existují distribuční systémy jedno stupňové a více stupňové. Ve více stupňovém distribučním systému mohou kromě finálního výrobce, zákazníka a dopravce být začleněni distributor, CrossDock, velkoobchod a maloobchod.



Obr. 1.1 Struktura distribučního systému

Zdroj: vlastní zpracování.

V případě krejčovské dílny a oděvní značky MAXFRED, která v současné době vyrábí striktně na zakázku, je současný distribuční systém jednostupňový, tedy bez jakéhokoli zprostředkovatele. To by se díky e-shopu sice změnilo, ale tento distribuční stupeň by byl pod kontrolou této značky, tudíž by nemělo dojít ke snížení kvality služeb s výrobkem spojených. Zároveň by mělo dojít k vytvoření malosériové výroby, i když i ta bude stále řízena poptávkou. Půjde tedy o unifikovanou výrobu na zakázku. K rozhodnutí vytvořit pro značku e-shop bylo učiněno na základě SWOT analýzy vytvořené společně s vlastníkem této značky a které z hlediska požadavků vlastníka na tvář této značky pro případné spotřebitele vycházelo jako nejlepší řešení pro rozvoj značky.

1.1 Více stupňový distribuční systém

Ve více stupňovém distribučním systému se vyskytuje vždy mezi finálním výrobcem a zákazníkem či jinak spotřebitelem nějaký zprostředkovatel či více zprostředkovatelů. Jejich druhy byly vyjmenovány výše. Jedná-li se o maloobchod jde o nákup zboží za

účelem jeho prodeje konečnému spotřebiteli. Velkoobchodem míníme mezičlánek distribučního systému, který nakupuje ve velkém a neprodává koncovému zákazníkovi. [1] Distributorem je myšlen zprostředkovatel mezi finálním výrobcem a velkoobchodem či maloobchodem, který zboží nenakupuje a pouze prodej či nákup zprostředkovává. CrossDock je pak mezičlánkem, který zprostředkovává pro maloobchod kompletování jednotlivých objednávek z hromadných dodávek od dodavatelů tak, že zboží se v CrossDocku neskladuje, pouze se rozděljuje a kompletuje dle objednávek a hned zas CrossDock opouští. Jinou možnou variantou více stupňového distribučního systému může být začlenění jako mezi stupně e-shopu.

1.1.1 Maloobchod

Maloobchod může existovat bez prodejny a v takovém případě se může jednat buď třeba o automat na mince nebo o online zprostředkování prodeje, například výše zmíněný e-shop. Jedná-li se o maloobchod s prodejnou, může nabývat několika forem a to „*standardní prodejny, specializované prodejny, úzce specializované prodejny, diskontní prodejny, supermarkety, hypermarkety, hobbymarkety, obchodní centra a obchodní domy*“ [1, s.33].

1.1.2 Velkoobchod

Vztah mezi prodejci a nakupujícími v rámci velkoobchodu vystihuje zkratka B2B, která pochází z anglického Business to Business. Tento výraz vystihuje to, že velkoobchod zprostředkovává obchod mezi obchodními společnostmi. Velkoobchod může být klasickým, specializovaným, s plným rozsahem služeb nebo s limitovaným rozsahem služeb. Do posledně jmenovaného se řadí i velkoobchod označovaným termínem „Cash-and-Carry“.

1.1.3 Distributor

Distributorem může být buď makléř nebo agent. U agentů se jedná o dlouhodobější vztah mezi prodávajícím či nakupujícím než u makléřů. Znakem obou těchto forem mezičlánku je, že zboží nevlastní ani neskladují, pouze za provizi prodej nebo nákup zprostředkovávají.

1.1.4 CrossDock

Počátek této formy distribuce lze lokalizovat do Spojených států amerických do 30. let 20. století. [1] Jedná se vlastně jen o překládku z vozů do vozů, kdy do CrossDocku přichází od několika dodavatelů vychystané dodávky na jednotlivých paletách pro jednotlivé maloobchody a ty se překládají do vozů pro tyto jednotlivé maloobchody a od jednotlivých dodavatelů se zkompletují do těchto dodávek.

1.2 Přímý distribuční systém

Tento distribuční systém je používán převážně k distribuci výrobků vyráběných na zakázku, jako u značky, které se tato práce týká. Kromě tohoto případu se může přímý distribuční systém používat v oblasti B2B, kdy jde o distribuci materiálu s unikátním charakterem. [2]

1.3 Volba rozšíření distribučních kanálů firmy

V této práci se řeší distribuční kanál pro oděvní značku MAXFRED, což je designová krejčovská dílna.



Obr. 1.2 Iška Fišárková

Zdroj: [3].

1.3.1 Charakteristika značky

Značka MAXFRED jejíž vlastníkem je Iška Fišárková vznikla v roce 1994. Značka byla pojmenována podle jmen synů paní Fišárkové Maxmiliána a Bedřicha. Jedná se o oděvní značku, pod kterou Iška Fišárková slavila mnoho úspěchů, například titul výtvarník sezony z Brněnských veletrhů STYL a KABO v roce 1998 a opět dvakrát titul výtvarník sezony tentokrát z Prague Fashion Week v následujících letech. Od začátku se jedná značku budovanou jako symbol kvality a originality a řemeslné preciznosti.

1.3.2 Historie značky

Před vznikem značky MAXFRED používala pro svou tvorbu Iška Fišárková značku IFmode, která ovšem ve své době byla zaměnitelná, co se autora týče. A tak s prvním podnikatelským počinem Išky Fišárkové, jímž bylo otevření obchodu na pražské Malé straně a manufaktury na pánské košile ve východočeském Slavětíně nad Metují, vzniká značka MAXFRED. Výroba pánských košil pak probíhala zhruba dva roky.

Obchod i manufaktura se zavírá přibližně současně. I přesto, že je Iška Fišárková převážně umělec a řemeslník, věnovala tenkrát výraznou péči pátému P marketingového mixu a pro své luxusní pánské košile si nechala vytvořit designový obal. Po uzavření manufaktury na košile, se pod značkou MAXFRED věnuje Iška Fišárková zakázkové tvorbě a tvorbě módních kolekcí jak pro soukromé přehlídky, tak pro veletrhy či festivaly.

K podnikání v pojetí jiném než řemeslném, se pod značkou MAXFRED Iška Fišárková vrací v roce 2007, kdy společně s Denisou Deákovou zakládají MAXFRED s.r.o. se sídlem Praha 6 – Hradčany, Na Baště sv. Jiří 257/5, PSČ 160 00. Tato společnost pak zaniká v roce 2017. Iška Fišárková pokračuje v umělecké oděvní tvorbě pod značkou MAXFRED opět v zakázkové tvorbě v soukromém atelieru.

1.3.3 SWOT analýza a její charakteristika

Název SWOT je zkratkou anglických slov Strengths, Weaknesses, Opportunities, Threats. Jednotlivá slova pak znamenají následující Strengths = Silné stránky, Weaknesses = Slabé stránky, Opportunities = Příležitosti, Threats = Hrozby. [5] SWOT analýza patří mezi základní metody strategické analýzy, což umožňuje její integrovaný charakter získaných, sjednocených a vyhodnocených poznatků, které slouží ke generování alternativ dalšího rozvoje organizace.

1.3.4 Historie SWOT analýzy

Autorem SWOT analýzy je Albert Humphrey. Analýza vznikla v 60. - 70. letech 20. století na Stanfordově univerzitě jako projekt financovaný Fortune 500 (500 největších korporací v USA). Cílem bylo na základě analýzy nedostatků v stávajícím plánování těchto společností vytvořit pro ně nový systém řízení změn. Původně se metoda pro týmové plánování, kterou vytvořil Albert Humphrey v rámci své práce na Stanfordském výzkumném ústavu, jmenovala SOFT analýza a až později byla přepracována a pojmenována SWOT analýza.

1.3.5 Použití SWOT analýzy

Analýza spočívá v rozboru a hodnocení současného stavu organizace (vnitřní prostředí) a současné situace okolí organizace (vnější prostředí). Výstupem analýzy jsou poté podklady pro stanovení rozvojových směrů a aktivit, podnikových strategií a strategických cílů.

Tab. 1.1 Rozvržení SWOT analýzy

SWOT-analýza	Kladné stránku	Záporné stránky
Interní analýza	S (Strengths) Silné stránky Vývoj nových metod, které jsou vhodné pro rozvoj silných stránek společnosti (projektu)	W (Weaknesses) Slabé stránky Odstranění slabín pro vznik nových příležitostí
Externí analýza	O (Opportunities) Příležitosti Použití silných stránek pro zamezení hrozeb	T (Threats) Hrozby Vývoj strategií, díky nimž je možné omezit hrozby, ohrožující naše slabé stránky

Zdroj: vlastní zpracování.

Silné a slabé stránky patří k vnitřnímu prostředí a jsou určovány vnitřními faktory efektivnosti organizace ve všech důležitých oblastech. Příležitosti a hrozby zase patří mezi vnější prostředí, to je oblasti, které organizace nemůže sama ovlivnit neboli existují nezávisle na organizaci a jejím konání a ovlivňují tedy organizaci z prostředí mimo ni. Příležitosti a hrozby identifikujeme ve všech důležitých oblastech, které organizaci ovlivňují. Na analýzu příležitostí a hrozeb lze implementovat PESTLE analýzu neboli je hodnotit v těchto oblastech: P – politické, E – ekonomické, S – sociální, T – technologické, L – legislativní, E – ekologické. Nikdy nesmí dojít k tomu, aby jeden identifikovaný faktor byl zároveň silnou a slabou stránkou nebo příležitostí a hrozbou. Smysl SWOT analýzy je naplněn pouze v případě, že získané výsledky budeme dále využívat, a to v podobě navazujících opatření, které vyplývají z definovaných strategií.

1.3.6 Členění SWOT analýzy

S jako strengths či silné stránky je část analýzy vnitřního prostředí. Silné stránky nám pomáhají posilovat tržní pozici neboli nám pomáhají identifikovat po kterých stránkách je naše organizace napřed před konkurencí. Cílem organizace je maximalizovat silné stránky nebo též vytěžit co nejvíce z toho co už umíme.

Do silných stránek patří různé dosažené úspěchy, dovednosti, znalosti, schopnosti nebo zdroje. Silné stránky jsou ovšem pouze to, co se vymyká z průměru. Například pro učitele angličtiny není silná stránka znalost angličtiny. Spíše je to nutný standard pro konkurenceschopnost.

W jako Weaknesses neboli slabé stránky je vnitřní (interní) část analýzy prostředí firmy. Obvykle jsou měřeny interním systémem hodnocení, benchmarkingem nebo srovnáním s konkurencí. Weaknesses jsou slabé stránky firmy, které snižují její vnitřní hodnotu (dovednost, aktiva, podnikové zdroje atd.). Cílem firmy je minimalizace slabých stránek.

O jako oportunities neboli příležitosti jsou potenciální vnější příležitosti – šance pro rozvoj podniku a posílení pozice na trhu. Podnik se proto snaží o jejich správnou identifikaci a maximální využití. Tyto skutečnosti mohou podniku přinést úspěch v podobě například větší poptávky, lepšího uspokojení zákazníků nebo zvýšení povědomí.

T jako threats neboli hrozby jsou faktory vnějšího prostředí a leží mimo kontrolu společnosti. Hrozby představují možná rizika, která ohrožují dosažení cílů společnosti

nebo její samotnou existenci. Cílem analýzy je hrozby identifikovat, sledovat a případně eliminovat.

Hrozby musí společnost systematicky dlouhodobě řídit a předcházet jim. V opačném případě mohou přerůst v závažný problém. Při řízení hrozeb je možné využít tzv. Paretova pravidla 80/20, kdy 20 % potenciálních hrozeb způsobí 80 % finančních ztrát. Při řízení hrozeb je proto nutné uspořádat jednotlivé položky postupně od nejzávažnějších po nejméně závažné.

1.3.7 Výstupy SWOT analýzy

Výstupem kompletní SWOT analýzy je chování společnosti, která maximalizuje přednosti a příležitosti a minimalizuje své nedostatky a hrozby. Vyhodnotí-li analýza, že převažují silné stránky a příležitosti, měl by se respondent ubírat po dobré cestě. Vyjde-li více slabin a hrozeb, je na zváženu, zda pokračovat. Pokud se respondent rozhodne jít dál, může si na základě SWOT analýzy stanovit strategické kroky, kterými se chce zabývat a které mu pomohou využívat příležitosti, eliminovat hrozby a slabé stránky.

1.3.8 Vypracování SWOT analýzy

SILNÉ STRÁNKY (Strengths)

- nadčasový design,
- špičkové řemeslné zpracování,
- osobní přístup ke klientovi,
- profesionálně vybavený ateliér,
- nezávislá výroba,
- exkluzivita,
- výborná dodavatelská tuzemská základna,
- opakované úspěchy v oblasti módního designu,
- zpracovávání přírodních materiálů,
- diverzifikace,
 - kurzy designu a šití pro laickou veřejnost,
 - designové poradenství,
- rodinná značka.

SLABÉ STRÁNKY (Weaknesses)

- cena výrobku,
- malý okruh klientely,
- nedostatečná propagace (PR – Public Relations),
- vysoké náklady,
- pouze jeden distribuční kanál,
- vysoké požadavky na distributory.

PŘÍLEŽITOSTI (Opportunities)

- Trend slow fashion (udržitelná móda),
- důraz na ekologii.

HROZBY (Threats)

- devalvace výroby a ceny oděvního výrobku,
- oděvní nad produkce,
- nedostatečný kulturní vyspělost obyvatelstva,
- zbytnost designu.

1.3.9 Výsledek SWOT analýzy

S ohledem na zájmy vlastníka značky, ohledně nároků na jakost produkce a řemeslné zpracování výrobků, je výstupem analýzy strategický cíl zlepšení propagace a vybudování nového distribučního kanálu. Cíl v propagaci je vytvořit designovou webovou stránku s informacemi o dění kolem značky. Cílem pro nový distribuční kanál je vytvoření katalogu a možnosti objednání zboží čili vytvoření e-shopu. Jiné cíle jsou s ohledem na povahu podnikání pro vlastníka nerelevantní.

2 Informační podpora malé firmy

Pro digitální podporu značky MAXFRED je možno, jak již bylo specifikováno ve výsledku SWOT analýzy, udělat několik kroků. Jedním z těchto kroků je zlepšení propagace v digitálním prostředí a druhým je rozšíření distribučních kanálů o e-shop. Pro propagaci v digitálním prostředí se používá digitální marketing.

2.1 Digitální marketing

Digitální marketing se též nazývá online marketingem a patří mezi čtvrté P marketingového mixu (Promotion), zatímco vytvoření nového distribučního kanálu pro značku spadá pod třetí P marketingového mixu (Place). Prvním P marketingového mixu je pak výrobek (Product) a druhým cena (Price). Online marketing má svá specifika, jako je možnost okamžité zpětné reakce zákazníka nebo též médium, na kterém je provozován. Médium může být jakákoliv technologie, založená na bitové technologii neboli je provozován na nějakém elektronickém médiu. Jednodušší, než definovat co přesně je digitální marketing, je naopak definovat co digitální marketing není. Digitálním marketingem nejsou tradiční média pro marketingovou komunikaci jako je rádio, televize, tisk nebo například billboardy. [6]

2.1.1 Historie digitálního marketingu

První náznaky online marketingu sahají do 90 let 20.století. Konkrétně v roce 1994 se objevuje první internetová reklama. [6] Tou dobou začíná být vedoucím pracovníkům některých firem zřejmý potenciál internetu, i když ještě dlouhou dobu je díky omezenému přístupu lidí k internetu nevyužitý. Postupem času se ale na internet začíná umisťovat propagace, nejdříve pouze jako brožury, postupem času jako katalogy, až nakonec se do digitálního prostředí přesouvá i samotný prodej produktů.

Časem se z jednoduché propagace staly složité procesy založené na zákaznické preferenci a chování na síti, názorech a připomínkách a vznikl tak internetový marketing. V současnosti již některé firmy zcela opouští od tradiční formy offline marketingu a jen čas ukáže, zda je to krok správnou stranou.

V roce 2003 pak začínají důležitou roli v online marketingu hrát nově se objevující sociální sítě.

2.1.2 Význam digitálního marketingu

Je nezpochybnitelné, že význam online marketingu bude růst. Význam online prostředí se obzvláště projevil v současné době karanténních opatření spojených s nemocí covid-19. Již při první vlně epidemie, dle vyjádření zaměstnance jednoho nejmenovaného e-shopu, byly obraty této společnosti na úrovni Vánoc. A je velmi pravděpodobné, že mnoho zákazníků, kteří si zvyknou nakupovat přes internet, bude tuto variantu upřednostňovat i po zklidnění situace.

Mimo to umožňuje online marketing propagaci nenásilnou formou, takže má zákazník možnost volby, zda se nechá upoutat a oslovit. Což často zjednodušují výhody internetového marketingu jako je zacílení reklamy na zákazníka. Důležitými výhodami online marketingu oproti tradičním médiím je možnost pružně měnit nabídku, schopnost osobního přístupu, nepřetržitá dostupnost, šířkou škály možností oslovit a v neposlední řadě možností zaznamenávat a měřit.

2.1.3 Výhody digitálního marketingu

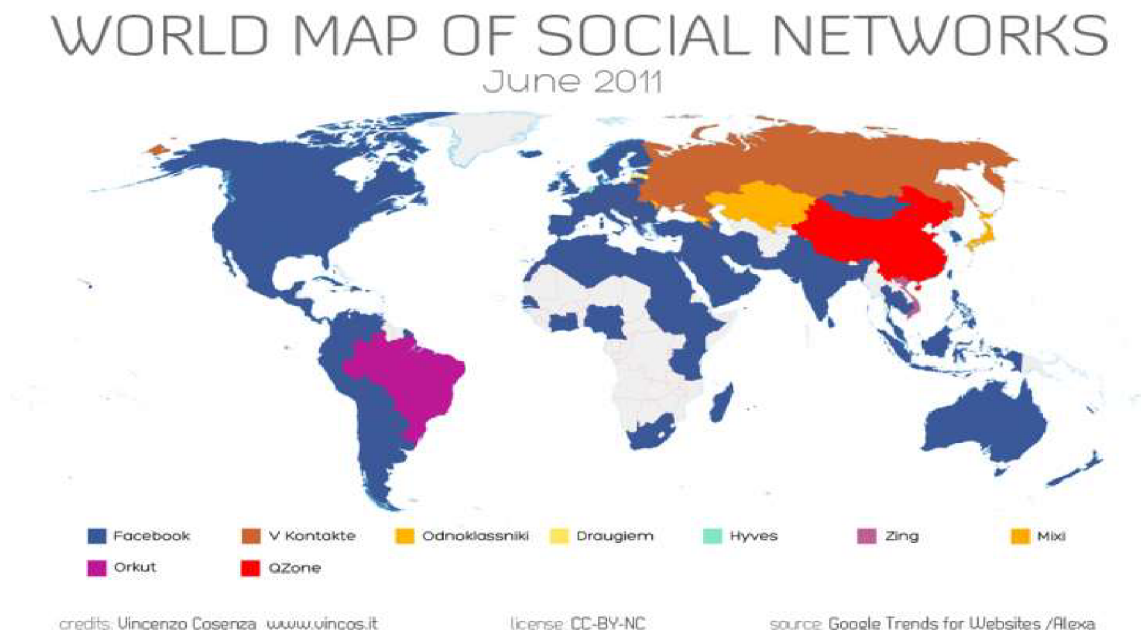
Mezi hlavní a nespornou výhodou se počítá možnost okamžitého hodnocení výsledků a dopadů online marketingové kampaně. Díky tomu je možné kampaň porovnávat s tradičními formami marketingu a hodnotit tak její přínos. V online prostředí je také možná nepřetržitá podpora v souvislosti s produkty, což zvyšuje spojenost zákazníků. Využívání sociálních médií pak umožňuje přijímání zpětné vazby ať pozitivní či negativní a pomáhá zjistit, které online komunikační kanály jsou nejatraktivnější. Mimo komunikaci ve směru firma-zákazník, umožňují sociální média také komunikaci mezi spotřebiteli, kteří zde mohou zveřejňovat zpětnou vazbu ohledně produktu či firmy. Tento způsob propagace se nazývá Word-of-mouth marketing a jeho význam je překladem tohoto anglického názvu neboli ústní sdělení informací o předmětu komunikace mezi různými skupinami obyvatel.

2.1.4 Sociální média

Důležitým prvkem marketingu je PR (Public Relations), mezi jehož základy patří komunikace se zákazníkem. V posledních 50 letech prošla komunikace obrovským přerodem. Na počátku tohoto přerodu, když jsme chtěli někomu předat nějakou informaci tak jsme mu poslali dopis, o čtvrt století později jsem měli již možnost mu zavolat pomocí telefonu třeba domů. Po roce 2000 již začalo být běžné, že jste mohli někomu zavolat více méně kamkoliv díky mobilním telefonům, před nedávnem začalo být běžné, že jste někomu poslal e-mailovou zprávu a dnes je nejběžnější, že někomu zanecháte zprávu na sociální síti.

Sociální média jsou fenoménem současné doby a jsou velmi dobrým prostředkem pro marketing. Nejde sice o bezprostřední prodejní kanál, ale jde o vynikající způsob, jak se dostat do povědomí zákazníků. Sociální media patří mezi nejrozšířenější prostředky internetového marketingu. Důležitým momentem při užívání sociálních médií pro marketing je dovednost řešit krizové situace a negativní ohlasy. Existuje několik různých druhů sociálních médií a to blogy, diskusní fóra nebo sociální sítě.

V současnosti jsou mezi sociálními sítěmi dominantní hlavně Facebook, Twitter, Google+, LinkedIn, Instagram, v Rusku VKontaktě a v Číně Tiktok.

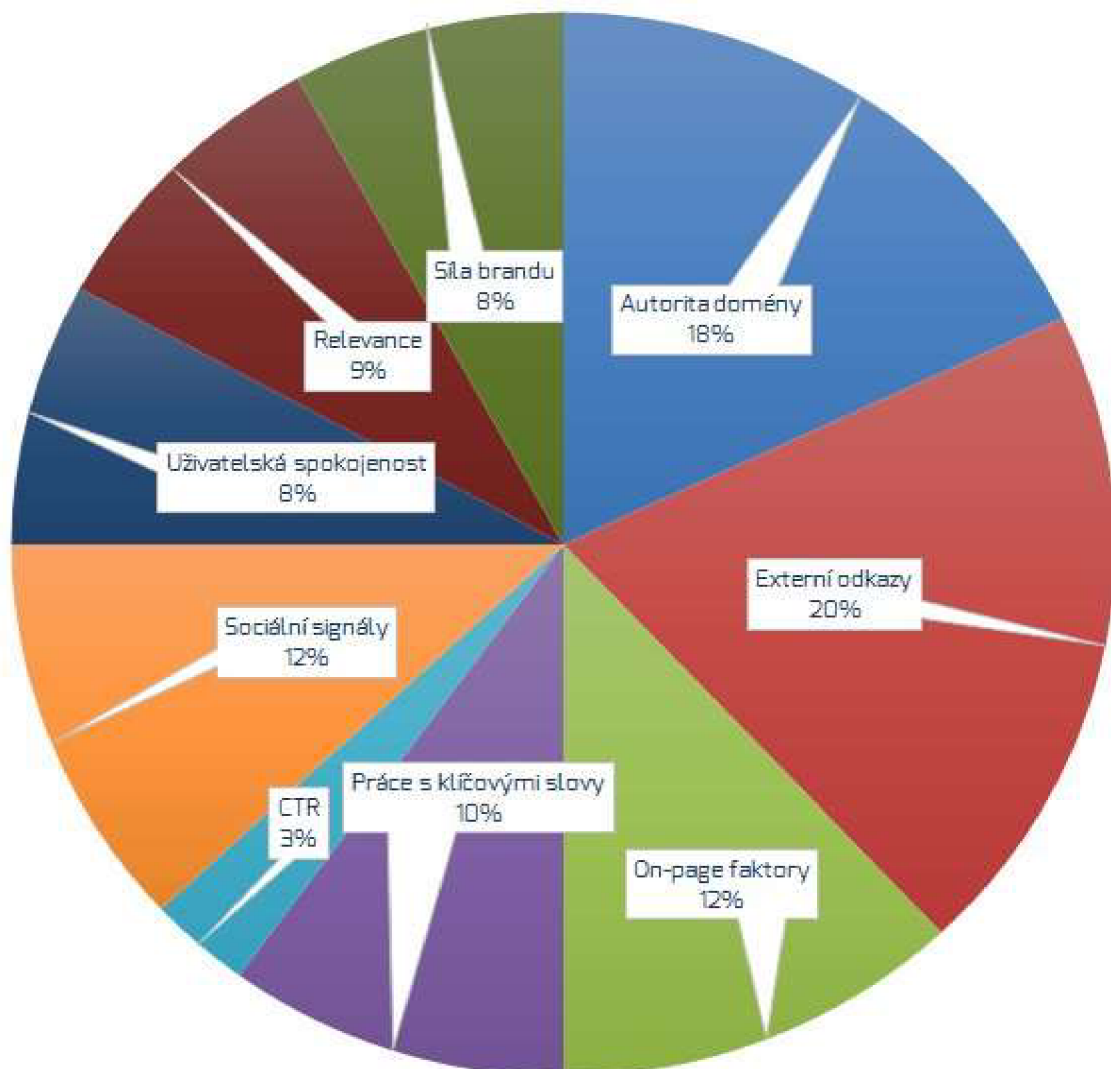


Obr. 2.1 Mapa sociální sítí ve světě

Zdroj: [7].

2.2 SEO – Search Engine Optimization

Nejlépe asi popisuje význam tohoto anglického výrazu název optimalizace nalezitelnosti. Jde o postup úpravy webu tak, aby se v neplacené části výsledků vyhledávání objevoval na předních pozicích, což samozřejmě má za následek častější návštěvnost vašich stránek. Vyhodnocování vhodnosti jednotlivých stránek k vašemu vyhledávání provádí prohlížeč a je založeno na nepřímých znacích jako jsou metadata, počty odkazů, struktura a podobně. Metod SEO je mnoho, jedná se o každou metodu, která pomáhá zviditelnit vaši stránku ve vyhledávačích. Cíle a metody pro lokálně dominantní vyhledávače by mohli být znázorněny níže.



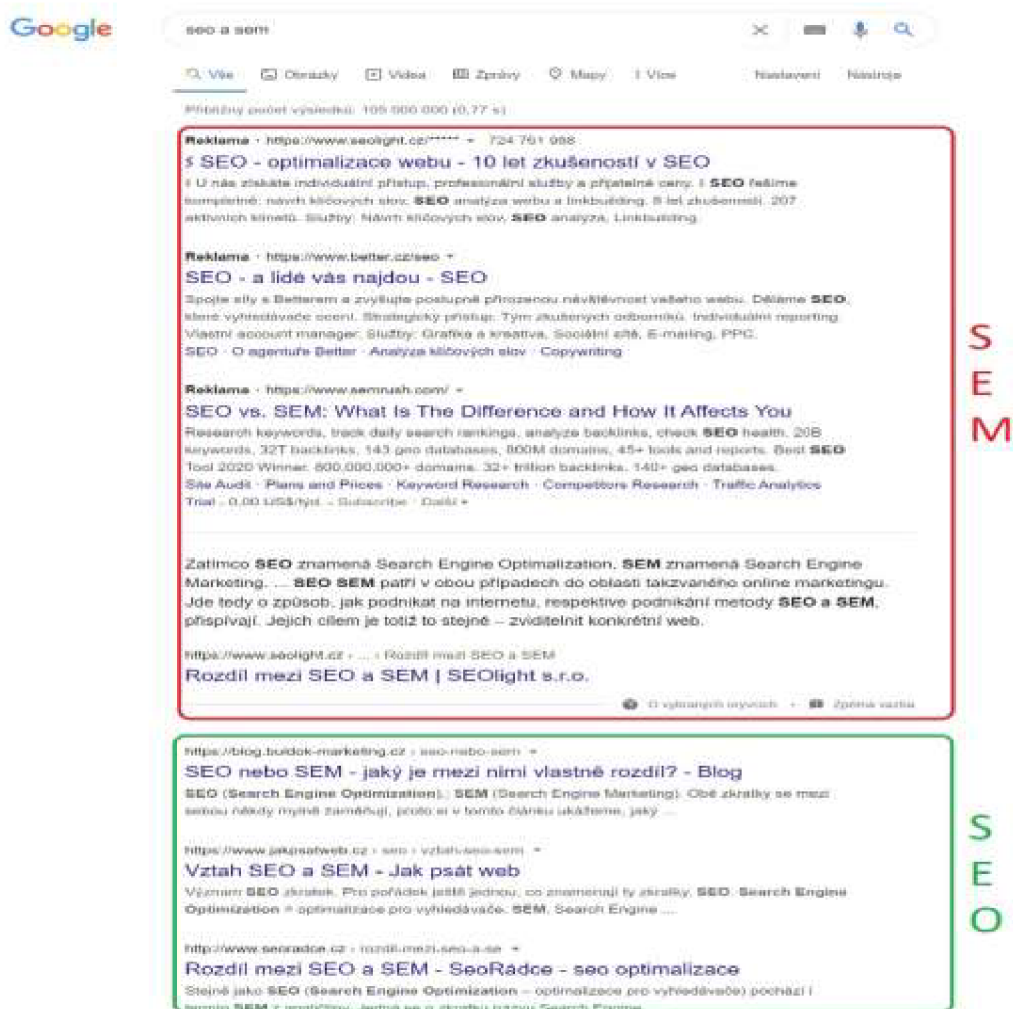
Obr. 2.2 Metody a cíle SEO

Zdroj: [8].

2.2.1 Etika v SEO

Etika s SEO je důležitá věc a v současnosti jsou již neetické SEO techniky prohlížeci vyhodnocovány jako důvod pro blokování zobrazení odkazu na vaše stránky. V základu se uvádí, že veškeré vámi provozované SEO techniky by měli být takové, které byste pro zlepšení stránek provedli i bez toho, abyste chtěli ovlivnit vaši pozici ve vyhledávači. Mezi tyto metody tedy patří primárně kvalitní a jedinečný obsah, krátká a stálá URL (Uniform Resource Locator – webová adresa), správné užití HTML (HyperText Markup Language) nebo aktivity na sociálních sítích.

Naopak mezi neetické metody a zároveň tedy riziko zablokování nalezitelnosti ve vyhledávacích patří spam, klamání vyhledávacího algoritmu anebo třeba skrytý text.



Obr. 2.3 Google SEO a SEM

Zdroj: vlastní zpracování.

2.2.2 SEM a rozdíly od SEO

SEM vychází z anglického názvu Search Engine Marketing a je s SEO velmi provázána. Hlavním společným znakem obou technik je společný cíl. Rozdílem je pak různé užití technik k dosažení tohoto společného cíle. Kombinování obou přístupů je velmi účinné. Užití SEO jsou interní a jednorázové náklady, tedy pokud není nějaká část outsourcována, zatím co SEM vyžaduje pravidelnou platbu externímu zprostředkovateli reklamního systému a je závislá na množství finančních prostředků do této techniky investovaných.

Výhodou SEM oproti SEO je rychlost, se kterou se projevují výsledky při dosahování jejich společného cíle. Zatím co u technik SEO se výsledky projevují po uplynutí většího časového úseku v rámci měsíců, u technik SEM, se výsledek projevuje prakticky okamžitě.

2.3 Východiska pro značku

Protože se v případě značky MAXFRED jedná o malou společnost bez vlastních specializovaných zaměstnanců, bude pro ni nejlepším východiskem k užívání metod digitálního marketingu hlavně propagace na sociální síti. Značka nemá svého marketéra, který by se věnoval budování profilu v online prostředí a půjde tedy o doplňkovou činnost vlastníka značky paní Fišárkové. Samozřejmě bude vhodné do aplikace e-shopu zapracovat podle metod SEO, třeba názvy URL, pro lepší zobrazování ve výsledcích vyhledávání. Techniky SEM jsou vzhledem na velikost společnosti nevhodné, a tedy nebudou pro zlepšení nalezitelnosti ve vyhledávacích používány.

3 E-shop a jeho realizace

Při realizaci samotného e-shopu bude třeba řešit několik problémů, které jsou s vývojem softwaru spojeny. A to je vytvořit si smysluplnou specifikaci cílového produktu a zvolit vhodný postup pro dosažení optimálního výstupu. Zároveň bude nutné zvažovat prostředky vhodné pro tvůrce, tím je míněno vhodný programovací jazyk a vývojové prostředí.

3.1 Požadavky na konkrétní řešení

Protože budeme potřebovat e-shop s menším rozsahem katalogu zboží a kategorií a bez zboží na skladě neboli budeme potřebovat místo kontroly stavu skladu, kontrolu kapacity dílny, je tedy potřeba vytvořit řešení odpovídající těmto podmínkám. V současnosti je běžným postupem pro vývoj aplikací použití a modifikace takzvaného open source řešení, tedy vyhnutí se zdlouhavému programování celé aplikace od začátku. Dalším požadavkem, který je na aplikaci kladen, tentokrát ne ze strany logiky aplikace, ale z technické stránky provedení je, aby byla psána v jazyce C# v MVC architektuře, a to z důvodu znalosti technologie ze strany mé jako vývojáře. Výsledkem tedy je, že cílem bude najít nebo vytvořit řešení, které bude umožňovat nabídku zboží s výrobou na objednávku s kontrolou kapacity dílny a psanou v jazyce C# v MVC architektuře.

3.2 Jazyk C#

Jazyk C# je vysokoúrovňový objektově orientovaný programovací jazyk. Byl vyvinut firmou Microsoft a patří do platformy .NET Framework. Jazyk C# byl založen na jazycích C++ a Java (a vychází tedy i nepřímě z jazyka C). Lze jej užít k psaní webových aplikací, jako v mém případě, formulářových aplikací systému Windows, aplikací pro mobilní zařízení jako je PDA (Personal Digital Assistant) a mobilní telefony anebo databázových aplikací. Název pro jazyk C# vznikl z hudební notace, kde křížek (#) značí zvýšení noty o půl tónu neboli by šlo o notu cis. Podobným způsobem vznikl i název jazyka C++ který v syntaxi jazyka C znamená zvýšení proměnné C o jednu. Vysokoúrovňový programovací jazyk (jinak též vyšší programovací jazyk) je označení pro programovací jazyk s větší mírou abstrakce. Čímž je myšleno přiblížení zápisu zdrojového kódu

programu v určitém programovacím jazyce tomu, jak nad problémem uvažuje člověk. Opakem je nižší programovací jazyk, který se svým zápisem blíží tomu, jak po technické stránce funguje počítač (konkrétně procesor). Vyšší programovací jazyk by tedy měl být člověku lépe srozumitelný oproti nižšímu programovacímu jazyku. Zápis programu ve vyšším programovacím jazyku je obvykle kratší a lépe čitelný než program v nižším programovacím jazyce. To šetří čas programátora a snižuje pravděpodobnost výskytu chyb. Ve vyšších programovacích jazycích se používají prvky přirozeného jazyka a struktura kódu je logická. Další výhodou může být přenositelnost takže program po malých či žádných změnách může být spuštěn na rozdílných platformách. Nevýhodou je, že se musí kód pomocí kompilátoru překládat do nejnižších programovacích jazyků, do takzvaného jazyka symbolických adres, aby mu počítač rozuměl. Prvními vyššími programovacími jazyky byl Plankalkül vytvořený Konradem Zusem a Fortran z roku 1957. Vyšší programovací jazyky se dělí na procedurální neboli imperativní a neprocedurální jinak deklarativní. Dále pak se procedurální jazyky dělí na strukturované a objektově orientované a neprocedurální se dělí na funkcionální a logické. Jako příklad vyšších programovacích jazyků si uvedme Python, Perl, PHP, Java a C#.

```
0310 CIOV = $E456 ;CIO ENTRY VECTOR
0320 RUNAD = $02E0 ;RUN ADDRESS
0330 EOL = $9B ;END OF LINE
0340 ;
0350 ; SETUP FOR CIO
0360 ; -----
0370 *= $0600
0380 START LDX #0 ;IOCB 0
0390 LDA #PUTREC ;WANT OUTPUT
0400 STA ICCOM,X ;ISSUE CMD
0410 LDA #MSG&255 ;LOW BYTE OF MSG
0420 STA ICBAL,X ; INTO ICBAL
0430 LDA #MSG/256 ;HIGH BYTE
0440 STA ICBAH,X ; INTO ICBAH
0450 LDA #0 ;LENGTH OF MSG
0460 STA ICBLH,X ; HIGH BYTE
0470 LDA #$FF ;255 CHAR LENGTH
0480 STA ICBLL,X ; LOW BYTE
```

Obr. 3.1 Nižší programovací jazyk (Assembler)

Zdroj: [9].

```
import time
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
from matplotlib.legend_handler import HandlerLine2D
from matplotlib.backends.backend_pdf import PdfPages

...

# Vyčisti datovou sadu od prázdných položek
def clean_dataset(df):
    assert isinstance(df, pd.DataFrame), "df needs to be a pd.DataFrame"
    df.dropna(inplace=True)
    indices_to_keep = ~df.isin([np.nan, np.inf, -np.inf]).any(1)
    return df[indices_to_keep].astype(np.float64)

# Vykresli výsledky na graf - ulož ho do obrázku png a pdf souboru
def draw_plot(estis, trainresults, test_results, xlabel="x", ylabel="y"):
    xn = range(len(estis))
    line1, = plt.plot(xn, trainresults, 'b', label="Train")
    line2, = plt.plot(xn, test_results, 'r', label="test")
    plt.legend(handler_map={line1: HandlerLine2D(numpoints=2)})
    plt.ylabel(ylabel)
    plt.xlabel(xlabel)
    plt.xticks(xn, estis)
    annot_max(xn, test_results, plt, estis)
    plt.savefig("_".join(["fig", xlabel, ylabel, str(time.time()), ".png"])

    pp = PdfPages('multipage.pdf')
    plt.savefig(pp, format='pdf')
    pp.close()
```

Obr. 3.2 Vyšší programovací jazyk (Phyton)

Zdroj: [10].

3.2.1 .NET framework

.NET („dotnet“ vychází v překladu z tečka net jako network neboli síť) je název pro platformu zastřešující soubor softwarových technologií pro web, Windows a pocket PC (Personal computer). CLI (Common Language Infrastructure) je specifikace jádra pro .NET. Základní komponentou platformy je Microsoft .NET framework. Vývoj .NET aplikací je pak umožněn v Microsoft Visual Studio .NET Platforma nepřikazuje použití žádného konkrétního programovacího jazyka. Aniž by záleželo na jazyku, ve kterém byla aplikace původně napsána, se aplikace vždy přeloží do mezijazyka Common Intermediate Language. Nejčastějšími programovacími jazyky používanými pro vývoj .NET aplikací jsou VisualBasic, .NET a C#. Dalšími užívanými programovacími jazyky jsou C++/CLI, F# (funkcionální programovací jazyk), J# (jazyky podobný jazyku Java), IronPhyton, Object Pascal, Boo.

3.3 MVC architektura

MVC je zkratka z anglického Model-View-Controller a jedná se o oblíbený architektonický vzor používaný převážně na webu. Původně vznikl pro desktopové aplikace v 70. letech minulého století. Je součástí oblíbených webových frameworků, jako jsou například Ruby On Rail pro Ruby, Zend či Nette pro PHP nebo MVC pro ASP .NET (ASP – Active Server Pages). Celkově je složitější web bez MVC architektury (nebo nějaké podobné architektury) jen těžko představitelný. Základním principem MVC architektury je oddělení logiky aplikace od výstupu pro uživatele. Pomáhá řešit problém tzv. „špagetového kódu“, kdy se v jednom souboru (třídě) střídá kód logických operací a zároveň renderování výstupu. Soubor tedy obsahuje jak databázové dotazy, logiku a různě rozesté HTML tagy, všechno je pak promíchané jako špagety na talíři. Takový kód je jen těžko udržovatelný, a ještě hůře rozšiřovatelný. Naším zájmem je mít zdrojový kód obsahující logiku vypadající jako čistý kód (například C#) a výstup jako HTML stránka nebo v našem případě Razor. Celá aplikace je tedy rozdělena do tří komponent, které jsem již zmínil výše, jedná se o Modely, View (pohledy) a Controllery. Model je datová struktura a logika, může obsahovat výpočty, databázové dotazy nebo třeba validace. Model vůbec nezná nic o výstupu. Jeho funkce je v přijetí parametrů a vydání dat ven a neví odkud data v parametrech pochází a jak budou výstupní data zformátována a zobrazena. View (pohled) se stará o zobrazení dat koncovému uživateli. Nejčastěji se jedná o nějakou HTML šablonu s tagy nějakého značkovacího jazyka, který umožňuje vkládat proměnné, provádět cykly a podmínky. Šablony lze samozřejmě začleňovat do sebe, abychom se zbytečně neopakovali. Například v šabloně s layoutem stránky je začleněna šablona s menu a v ní pak ještě šablona s konkrétní nabídkou. View pak podobně jako Model netuší odkud data, která zobrazuje přišla, stará se pouze o jejich zobrazení. Controller je spojovacím prvkem, který zprovozňuje celou architekturu. Jde o jakého si prostředníka, s kterým komunikuje View, Model i uživatel. Životní cyklus stránky pak vypadá přibližně takto. Životní cyklus zahájí uživatel zadáním adresy webu a parametry. Požadavek zpracovává tzv. Router a ten dle parametrů pozná, který controller voláme a předá mu zbylé parametry. Konkrétní controller dle parametrů pozná, co je po něm vyžadováno a zavolá model. Model vyhledá data v databázi a případně s nimi provede nějakou operaci, třeba vytvoří ze jména a příjmení celé jméno a pak je odešle controlleru. Tyto data controller vloží do proměnných, předá je view, který je určen podle probíhající operace a ten konečně vygeneruje pohled neboli stránku, která se

zobrazí uživateli. Controller tedy zpracoval požadavek od uživatele, obstaral data od Modelu a předal je View. View data přijal a vložil do připravené šablony, která se pak zobrazila uživateli.

3.4 Open Source versus vlastní zdrojový kód

Otázkou kladenou si už v úvodu bylo, zda použít vlastní zdrojový kód nebo si propůjčit řešení, které je na internetu nabízeno volně k užití, a to si následně upravit pro vlastní potřeby.

3.4.1 Open Source

Otevřený software (anglicky open-source software) je takový počítačový software, jehož zdrojový kód je otevřený. To že je zdrojový kód otevřený znamená technickou dostupnost kódu a legální dostupnost (nedostupnost) softwarové licence. Licence je v tomto případě velmi důležitá, protože specifikuje, která práva k otevřenému kódu získává a jak s nimi může uživatel nakládat. Existují licence (například GNU – General Public License), které umožňují uživatelům zdrojový kód upravovat a distribuovat. V posledních letech se open source stal podle analytiků zdrojem inovací v oblasti software a převzal tak štafetu od proprietárního softwaru. Vzrostl také zájem firem o open source, obzvláště v oblasti webových řešení a také se rozostřily hranice mezi softwarem, který je a který není open source. Příbuzným tématem je svobodný software (anglicky free software) – který jako pojem prosazuje Free Software Foundation již od 80. let 20. století. Následně pak v roce 1998 lidé, kteří se pokoušeli prosadit Free Software ve světě komerčních firem, přišli s novým názvem pro Free Software – Open Source. K této změně je vedla obava, že termín „Free Software“ komerční firmy mate a odrazuje. Tento nový název pak organizace OSI (Open Source Initiative) převzala a učinila z něj svůj oficiální termín, zatímco FSF (Free Software Foundation) zůstala u termínu Free Software. Odborníci v oboru softwaru identifikovali pro open source různé výhody a nevýhody. Nejdůležitější výhodou pro podnikání je to, že díky open source je možné dospět k lepšímu pronikání na trh. Společnosti poskytující open source software mohou stanovovat průmyslovou normu a díky tomu získat nad konkurencí výhodu. Pro OSS (open source software) nejsou zapotřebí takové náklady na marketing a logistické služby a vývoj vytváří spolehlivý a velmi kvalitní software levně a rychle. Dokonce by měl být daleko spolehlivější, protože

se na jeho vývoj podílejí tisíce nezávislých programátorů, kteří software testují a opravují v něm chyby. OSS je flexibilní, jelikož modulární systémy umožňují programátorům vlastní rozhraní či přidávání nových funkcí. Směs rozdílných pohledů, osobních a firemních cílů zrychluje inovace systémů. Mimo to free software může být vyvíjen ve shodě s technickými požadavky, neboť není pod komerčním nátlakem, který snižuje kvalitu softwaru. Také to, že vývojář není pod tlakem, působí pozitivně z hlediska času věnovaném zabezpečení namísto požadavků zákazníka. Občas se tvrdí, že vývojový proces OSS nemusí být správně definován a některé části, jako je například testování a dokumentace, bývají zanedbávány. To však platí jen pro malé (většinou jednoho programátora) projekty. U větších projektů je nutné stanovit a dodržovat jistá pravidla, aby byla možná jednoduchá týmová práce. Problém ovšem je navrhnout komerčně lákavý podnikatelský model dle open source předlohy. To znamená, že splněny mohou být jen technické požadavky a už ne požadavky trhu. Také pokud jde o bezpečnost, je u open source řešení pro hackery snazší se dozvědět o slabých místech a mezerách systému než u closed source softwaru. Mezi nejznámější open source řešení patří mezi operačními systémy GNU/Linux a Android, mezi databázovými servery MySQL a PostgreSQL, za webservery Apache, mezi kancelářským softwarem OpenOffice.org a LibreOffice, mezi internetovými prohlížeči Mozilla Firefox a Chromium, za vektorový editor Inkscape a za multimediální přehrávače třeba VLC media player nebo Mplayer.

3.4.2 Výhody vlastního kódu

Oproti open source řešení vlastní kód poskytuje jasnou představu o tom, jak aplikace vnitřně funguje. Jestliže k tomu ještě jsou dodržována pravidla pro strukturu kódu a třeba právě MVC architekturu, může být pak udržování kódu jednodušší.

3.5 Výstup analýzy prostředí

Pro e-shop pro rodinou firmu, o kterém je tato práce, jsem se nakonec na základě mnou získaných informací rozhodl pro vlastní řešení. Rozhodující pro mě bylo, aby open source řešení bylo určeno pro výrobu na poptávku se správou kapacity dílny a bylo v MVC ASP .NET. Takové jsem ovšem nenašel. Jediné open source psané pro výrobu na poptávku nebylo za prvé v MVC ASP .NET a za druhé jeho licence nebyla jednoznačně open source i pro komerční užití, které je u tohoto e-shopu předpokládáno. Jiná řešení, která byla psána

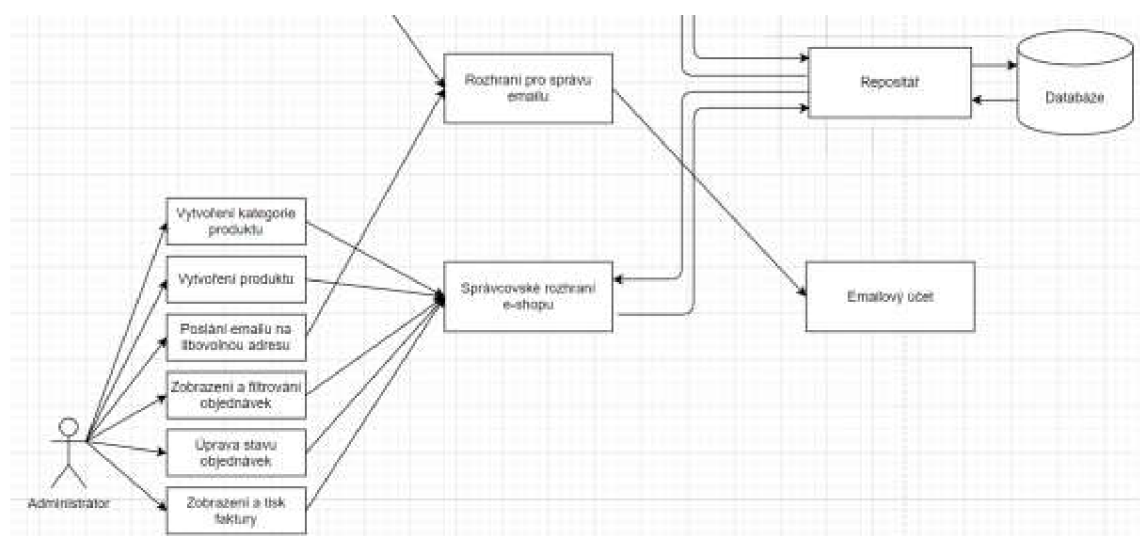
na platformě MVC ASP .NET zas neplnila podmínku výroby na poptávku a byla pro požadované řešení zbytečně složitá, což by v budoucnu po nutné úpravě kódu pro výrobu na poptávku mohlo znamenat složitou údržbu.

4 Návrh řešení

Pro napsání aplikace byly zvoleny tyto frameworky a programovací jazyky. Jako základ byl zvolen jazyk C# a platforma .NET a její framework ASP.NET v architektuře MVC. Pro psaní front-end části aplikace byl použit jazyk Razor, který je nativní součástí jazyka C#. Pro databázovou část aplikace byl zvolen Entity framework a jeho jazyk Linq (Language-Integrated Query).

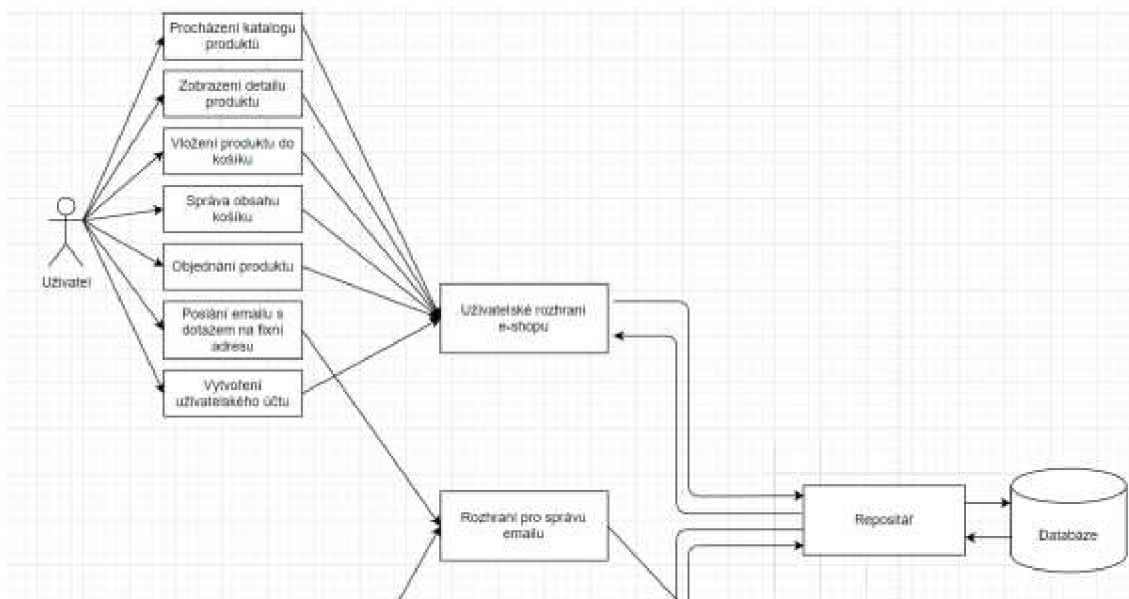
4.1 Postup při návrhu konceptu aplikace

Možnost použití open source řešení byla projednávána v předchozím textu a pro složitost dostupných řešení při požadovaných vlastnostech byla zavržena. Byla však zvolena možnost napsání základní struktury e-shopu podle tutoriálu k naprogramování takovéto aplikace. Tento tutoriál popisuje naprogramování e-shopu se stavem zboží na skladě a správou množství v něm, což nebylo ve výsledné aplikaci žádoucí, avšak díky postupnému vlastnímu psaní zdrojového kódu bylo snazší jej upravit. V průběhu procházení tutoriálem, jak napsat takovouto aplikaci, docházelo k situacím, kdy do vzorového zdrojového kódu byly vkládány metody, vlastnosti a třídy, které řešily specifika pro požadované konkrétní řešení.



Obr. 4.1 Schéma struktury správcovského rozhraní

Zdroj: vlastní zpracování.



Obr. 4.1 Schéma struktury uživatelského rozhraní

Zdroj: vlastní zpracování.

Hlavními body odlišnosti oproti tutoriálu bylo zvolení přístupu Code-first při navrhování databáze v Entity frameworku. Dalším bodem bylo řešení kapacity dílny místo stavu zboží na skladě. Třetím bodem odlišnosti byla možnost platby bankovním převodem oproti pouze platbou přes platební bránu Paypal. Dále byla do aplikace doplněna funkce rozesílání emailu s fakturou primárně v návaznosti na možnost platby bankovním převodem. Také byla do aplikace začleněna možnost poslat zprávu s dotazem na email e-shopu a v neposlední řadě byly u zboží řešeny velikosti a na stránky umístěna tabulka rozměrů odpovídajících konfekčním velikostem.

4.2 IDE – vývojové prostředí

Aplikace byla vyvíjena v IDE (integrated development environment) neboli vývojovém prostředí Visual Studio 2019 ve verzi Community Edition, která je dostupná i pro vývoj komerčních produktů bezplatně. Dalšími dvěma verzemi jsou verze Professional Edition a Enterprise Edition, které jsou ovšem již zpoplatněné.

Visual Studio 2019 je, co se týče komplexnosti a různých podpůrných funkcí pro vývoj aplikací rozhodně jedno z nejpokročilejších vývojových prostředí. Intellisense ve Visual Studiu umožňuje, nejen znázornit špatnou syntax nebo znázornit chybně zapsané klíčové slovo, ale též nabízí většinou všechny možné užití metod nad instancí. Dále třeba

znázorňuje a reportuje podmínku s neměnným výstupem, označuje definované proměnné bez následného použití v kódu a také navrhuje možné opravy nalezených chyb.

Visual Studio 2019 Community Edition je ke stažení přímo na stránkách firmy Microsoft je toto EDI vyvíjí. Konkrétně se jedná o URL: <https://visualstudio.microsoft.com/cs/downloads/>.

4.2.1 Nastavení a instalace Visual Studia 2019

Po stažení souboru pro instalaci Visual Studia se otevře takzvaný Visual Studio Instalér, který teprve začne stahovat a instalovat soubory. Po stažení a instalaci základních nezbytných komponent Instalér nabídne volbu pro instalaci frameworků pro vývoj různých typů aplikací. Pro vývoj webové aplikace jde o framework ASP.NET a web development, který obsahuje nezbytné komponenty pro vývoj webové aplikace. Po nainstalování Visual Studio vyžaduje registraci pod libovolným emailovým účtem.

4.2.2 Založení nového projektu

Možnost založit nový projekt Visual Studio nabízí v úvodním dialogu po spuštění, po zvolení této možnosti, je ještě třeba specifikovat druh aplikace, kterou chcete v projektu vyvíjet. V případě e-shopu pro značku MAXFRED byla zvolena ASP.NET Web Application (.NET framework), dále je nezbytné zvolit název projektu, umístění na disku a případně název řešení, což je kontejner, který může obsahovat více jak jeden projekt. Nakonec je možné zvolit verzi .NET frameworku, ve které bude projekt vyvíjen.

4.3 Databáze

Pro řešení databáze byl zvolen Entity Framework, který umožňuje pomocí jazyka Linq obsluhovat databázi přímo v kódu v jazyce C#. Alternativou k tomuto řešení je pro širokou veřejnost známější návrh a obsluha databáze v jazyce SQL (Structured Query Language).

4.3.1 Structured Query Language

SQL je jazyk pro správu a návrh relačních databází. První návrh specifikace pro relační databázi pochází od Edgara F. Codd z roku 1970, tehdy pracující jako zaměstnanec

společnosti IBM (International Business Machines Corporation). Předchůdcem jazyka SQL byl jazyk vyvinutý právě společností IBM s názvem SEQUEL (Structured English Query Language). Po IBM se vývojem relačních databází začala zabírat i firma Oracle Corporation (tehdejší Relational Software, Inc.). S příchodem dalších databázových systémů v počátku osmdesátých let, které také používali jazyk SEQUEL, se tento jazyk přejmenoval na SQL.

Příkazy v jazyce SQL se dělí v základu do čtyř skupin, a to na DML (Data Manipulation Language), DDL (Data Definition Language), DCL (Data Control Language) a TCC (Transaction Control Comands). Skupinu příkazů DML se česky překládá jako příkazy pro manipulaci s daty a jde o příkazy pro úpravu a získávání dat v databázi. Skupinu příkazů DDF je možno přeložit jako příkazy pro definici dat a jde o příkazy používané pro vytvoření struktury databáze. Poslední dvě skupiny příkazů DCL a TCC pak jsou nazývány jako příkazy pro řízení dat nebo konkrétně DCL jako příkazy pro ovládání dat a TCC jako příkazy pro ovládání transakcí.

4.3.2 Entity Framework

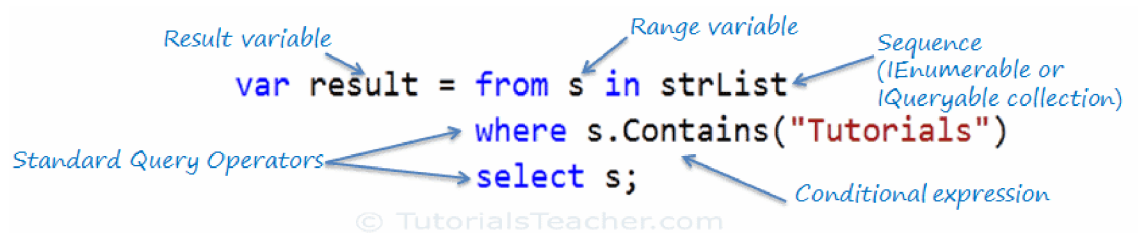
Entity Framework 6.0 je ORM framework (object-relation mapping), který byl použit pro vývoj e-shopu. Verze 6.0 byla první která byla osamostatněna od frameworku .NET a zároveň byla vydána jako open source. Zároveň framework od této verze umožňuje návrh databáze způsobem Code First.

Způsob návrhu databáze Code First umožňuje databázi navrhnout jako třídy a jejich vlastnosti. Vlastnost třídy, která typem jiná databázová třída, pak nahrazuje provázání pomocí cizího a primárního klíče pro relaci 1-1. Jde-li o vlastnost, která je kolekcí typu jiné databázové třídy, jde též o náhradu provázání pomocí cizího a primárního klíče, tentokrát však pro relaci 1-N. Vytvoření relace N-M je pak docíleno tak, že obě databázové třídy obsahují vlastnost kolekcí typu druhé databázové třídy.

Práce s Entity Frameworkem zjednodušuje přístup k datům v databázi po té stránce, že po dotazu na databázi se výsledek vrací jako instance třídy databázového modelu nebo kolekce těchto instancí. Tento přístup k datům v databázi zjednodušuje práci s nimi, protože není potřeba další formátování pro objektově orientované zpracování.

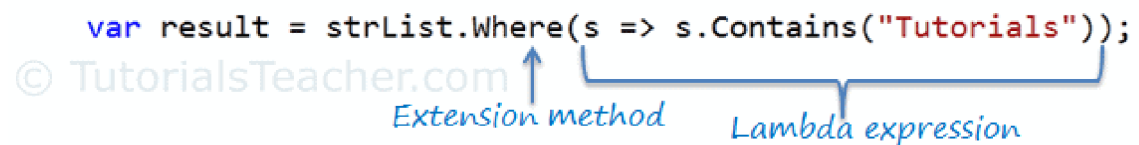
4.3.3 Linq

Language Integrated Query neboli LINQ je dotazovací jazyk pro práci s daty a je rozšířením jazyka C# nebo Visual Basic.Net. Jeho skladba dotazů vychází z jazyka SQL. Je součástí frameworku .NET od verze 3.5 tedy od roku 2007. Rozšířením jazyka LINQ, které bylo při vývoji e-shopu použito jsou takzvané lambda výrazy, které zjednodušují a zkracují zápis dotazu nad databází.



Obr. 4.3 LINQ query

Zdroj: [11].



Obr. 4.4 LINQ lambda query

Zdroj: [12].

4.3.4 Vytvoření a úprava databáze

Po navržení databáze způsobem Code First, je třeba Entity Frameworku zadat, aby vytvořil takzvanou migraci, která obsluhuje MS SQL server a vytvoří požadovanou strukturu databáze nebo ji upraví, jestliže jste provedli změny. Toto se vykonává v takzvaném Package Manager Console a musí se zvolit správný projekt, nad kterým se má vybraný příkaz vykonat, protože databáze se udržuje v samostatném projektu oddělena od zbytku aplikace pouze ve společném tzv. solution neboli řešení. Příkaz, kterým se databázová migrace vytvoří je Add-Migration, po vytvoření migrace je ještě třeba migraci nad databází spustit, a to pomocí příkazu Update-Database.

4.4 Správa uživatelských účtů

Pro správu účtů v aplikaci bylo použito zabudovaných tříd a metod včetně předdefinovaných UI (user interface). Uživatel je umožněno objednávku vykonat i bez zřízení účtu, s tím rozdílem, že registrovaný uživatel při opakované objednávce pouze kontroluje či upravuje, zda jsou údaje o jeho osobě a místě dodání objednávky v aktuálním stavu, zatím co neregistrovaný uživatel je musí pokaždé vyplnit znova.

Při žádosti o smazání účtu se odstraní uživatelský účet, fakturační údaje se pak odstraní v administrátorském rozhraní po uplynutí doby požadované pro uchování těchto údajů. Při registraci uživatelského účtu a při vkládání údajů o objednateli je uživatel vyzván, aby souhlasil s obchodními podmínkami.

V systému správy účtů existuje pouze jedna role a to role administrátor. Tato role je ověřována při pokusu o vstup do administrátorského rozhraní a pakliže se v uživatelské části e-shopu uživatel přihlásí pod administrátorským účtem, je mu pod volbami akcí s účtem nabízen přechod do administrátorského rozhraní.

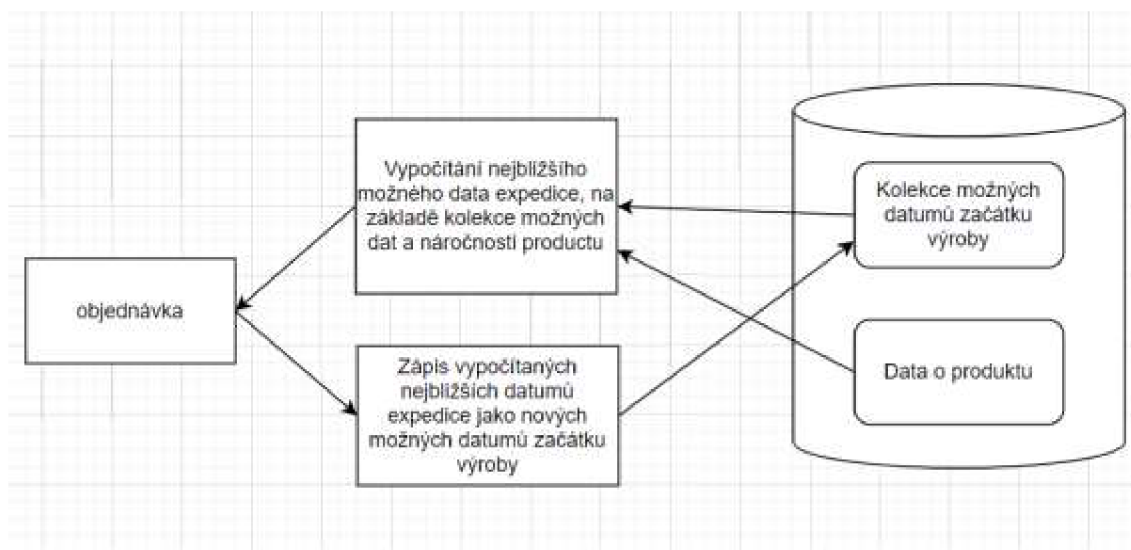
4.5 Řešení kapacity dílny

Kapacita dílny byla prvním rozdílem, co se týče provozní logiky mezi e-shopem psaným v tutoriálu a e-shopem pro značku MAXFRED. V e-shopu psaném v tutoriálu byl řešen stav zboží na skladě, což byla vlastnost instance třídy produkt, kdy se pro každý produkt v databázi držel počet dostupných kusů. V řešení pro MAXFRED tuto vlastnost u produktu nahradila vlastnost česky zvaná náročnost v modelu nazvaná „difficulty“. Tato vlastnost je celým číslem znamenající počet dní, které je potřeba pro ušití konkrétního produktu.

Dále byla vytvořeny třídy „Capacity“, „Delivery“ a „Deliveries“. Přičemž třída „Capacity“ obsahuje informaci o kapacitě dílny, to znamená, kolik produktů může zároveň dílna zpracovávat a třída „Deliveries“ obsahuje kolekci třídy „Delivery“ jež drží informaci o následujících datech, kdy se uvolní kapacita pro zpracování nového produktu. S vytvořením nové objednávky se pak do třídy „Deliveries“ zapíšou nová data pro budoucí objednávky.

4.5.1 Datum expedice

Pro zákazníka není až tak důležité, kdy se na jeho objednávce může začít pracovat, ale kdy celou svou objednávku dostane. Z tohoto důvodu aplikace před zobrazením možných dat expedice získá z databáze data možných začátků zpracování objednávky a na základě obsahu zákazníkovi objednávky kalkuluje nejbližší možné datum expedice. A to tak, že k nejbližšímu možnému datu začátku výroby nebo k současnému datu, pakliže první možné datum výroby je v minulosti, přičte obtížnost produktu. Počet uchovávaných možných dat začátku výroby pak odpovídá kapacitě dílny.



Obr. 4.5 Schéma výpočtu dat expedice

Zdroj: vlastní zpracování.

4.6 Řešení rozdílné velikosti produktů

Dalším rozdílem oproti vzorové aplikaci byla vlastnost produktu obsahující konfekční velikost. Jedná se vlastnost výčtového typu instance třídy produkt, který jednoduše umožňuje definovat hodnoty, kterých bude vlastnost nabývat. Při vkládání zboží do košíku se pak zkontroluje, zda se produkt neliší pouze druhem ale i velikostí, pakliže se liší velikostí, je vložen do košíku samostatně.

Druhým krokem při řešení velikostí produktů, bylo vložení .pdf (Portable Document Format) tabulky s rozměry konfekčních velikostí pro zpřesnění informace o velikostech. Tato tabulka je přístupná v uživatelské části e-shopu odkudkoliv pomocí záložky „Sizes“ nebo v detailu produktu z odkazu nad combo boxem pro výběr velikosti. V tabulce je

uživatel vyzván, ať v případě, že potřebuje produkt v individuálních velikostech, kontaktuje firmu.

4.7 Detail produktu

Detail produktu ve vzorové aplikaci nebyl vůbec řešen. V naší aplikaci byl požadavek, aby detail, oproti náhledové kartě, mohl obsahovat větší počet fotek produktu. Z toho důvodu byly do třídy produkt přidány vlastnosti typu string (neboli řetězec znaků) s názvem DetailImage1-5, které mohou obsahovat názvy fotek produktu, díky kterému je lze do detailu načíst.

Zobrazení většího počtu fotek produktu bylo řešeno pomocí „carouselu“, neboli pásu střídajících se fotek. Počet fotek v „carouselu“ byl dynamicky nastaven podle počtu fotek pro konkrétní produkt. Dále byl oproti náhledové kartě v detailu zobrazen popis produktu a kategorie produktu.

4.8 Správa objednávek

Ve vzorové aplikaci bylo řešeno vytváření a upravování kategorií a produktů, ne však správa provedených objednávek. Toto bylo tedy vyřešeno přidáním seznamu objednávek obsahující informace o datu expedice, číslu faktury, ceně objednávky a stavu objednávky z hlediska platby, zpracování a zrušení. Stav objednávky z hlediska platby je důležitý v momentu, kdy je rozdíl v platbě. V případě, že jde o objednávku placenou přes Paypal bránu, je objednávka zaplacená po úspěšném dokončení objednávky. V případě, že jde o objednávku placenou bankovním převodem, označuje objednávku za zaplacenou sám administrátor po přijetí platby s odpovídajícím variabilním symbolem. Zbylé dva stavy jsou po vytvoření objednávky vždy negativní a slouží administrátorovi k označení zpracovaných či stornovaných objednávek. Posledním prvkem seznamu objednávek je odkaz na detail objednávky.

Seznam objednávek obsahuje vyhledávací pole, do nějž lze například zadat číslo faktury pro vyhledání konkrétní objednávky, toto pole filtruje podle všech položek seznam, tudíž lze vyhledávat například i podle datu expedice nebo ceny.

Detail objednávky pak obsahuje veškeré informace o objednávce, jako je počet produktů a jejich druhů a velikostí nebo .pdf soubor s fakturou k objednávce, tak aby ji bylo možné

při expedování vytisknout a k zásilce přibalit. Detail objednávky také obsahuje odkaz na úpravu zakázky, kde je možné upravit údaje o stavu zakázky, jako je stav platby, zpracování nebo stornování a o datumu expedice.

4.9 Generování faktury

Generování faktury patří mezi další prvky, které vzorová aplikace neobsahovala. Pro možnost vygenerování .pdf souboru bylo nutné do projektu nahrát framework tuto činnost umožňující. Po průzkumu použitelných knihoven a na základě dostupnosti dokumentace o jejich používání, byla vybrána knihovna itext7. Knihovna itext7 určená pro .NET je dostupná pod licencí GNU Affero General Public License.

Pro samotné vytváření faktury byla v adresáři PDF, kam se též ukládají vygenerované .pdf soubory s fakturami, vytvořena třída pro zpracování dat z instance třídy databázového modelu objednávky.

Název .pdf souboru byl zvolen jako kombinace anglického výrazu pro fakturu a čísla faktury, jež je zase kombinací roku vystavení faktury a čtyř ciferního sufixu odpovídajícímu pořadovému číslu objednávky. Faktura byla vytvořena jako tabulka obsahující souhrnné informace o smluvních stranách a o obsahu objednávky.

Faktura

Faktura číslo		Datum Vystavení			
20210018		09.03.2021 0:00:00			
Dodavatel		Odbratel			
MAXFRED Matjuchinova 704 Praha 5 - Zbraslav 156 00		Maxmilian Fišárek Nová Šárka 511/7, 7 Praha 6 - esko 16100 email.:maxmilianleft@gmail.com			
IC: 49893267 DIC: 168765498794 C. uctu: 1654846546/1654 Banka: CSOB a.s.		forma úhrady: Invoice Variabilní symbol: 20210018			
Produkt	množství	cena v Kc bez DPH	sazba DPH v %	DPH v Kc	Kc celkem včetně DPH
Mazarin	2	7742,00	21 %	2058,00	9800,00
test1	2	4740,00	21 %	1260,00	6000,00
Celkem		12482,00		3318,00	15800,00
Celkem k úhradě					15800,00

Obr. 4.1 Vygenerovaná faktura

Zdroj: vlastní zpracování.

Cesta pro uložení souboru byla vytvořena pomocí vlastnosti `CurrentDomain.BaseDirectory` třídy `AppDomain`, což umožní nalezení cesty při libovolném nasazení aplikace bez nutnosti jejího nastavování. Stejně vlastnosti pak bylo použito při nastavování cesty ke vložení `.pdf` souboru faktury do emailu, odesílaného po potvrzení objednávky nebo při zobrazení faktury v rozhraní detailu objednávky. Při zobrazení přes odkaz v detailu objednávky je faktura otevřena v novém okně a je možné ji vytisknout, aby mohla být přibalena k zásilce.

Pro formátování textu uvnitř jedné buňky tabulky tvořící základní strukturu faktury bylo nezbytné použít regulárních výrazů.

4.10 Zasílání emailových zpráv

Další nezbytností při zpracování objednávky, kterou vzorová aplikace neobsahovala, bylo rozesílání emailové zprávy obsahující například výše zmíněnou fakturu nebo třeba též dotaz zákazníka. V tomto případě nebylo nutné instalovat jakoukoliv rozšiřující knihovnu, protože možnost správy emailové schránky v sobě má zabudovaný už `.NET` framework.

Pro odesílání emailových zpráv byl vytvořen samostatný kontroler, jehož jedna část je přístupná pouze po ověření, že je uživatel přihlášen jako administrátor, a druhá část, která je dostupná i neregistrovanému uživateli. Obě části obsluhují stejný emailový účet, pořízený právě pro chod e-shopu, avšak část přístupná i neregistrovanému uživateli neumožňuje nastavit cílovou adresu. Pro uživatele je cílová adresa pevně nastavena na emailovou adresu emailu e-shopu, tak aby z aplikace mohl odeslat dotaz. Obě rozhraní pro odesílání zpráv, ať již určené pro administrátora či uživatele, umožňují vkládání souborů. Autorizace umožňující přístup pouze do některých částí celé třídy kontroleru je řešena přímo u jednotlivých metod.

Jednotlivé metody jsou pak přístupné, přes odkazy v jednotlivých rozhraní, tedy v uživatelském rozhraní e-shopu a v administrátorském rozhraní. Pro vygenerování formuláře, byla použita CSS (Cascading Style Sheets), JS (JavaScript) a HTML knihovna

Bootstrap, která je určena pro vývoj responsivního pro mobilní telefony optimalizované webové stránky.

4.11 Platební brána Paypal

Posledním krokem, stěžejním pro fungování e-shopu, bylo začlenění Paypal platební brány. Toto bylo řešeno i ve vzorové aplikaci. K začlenění bylo třeba instalovat knihovnu Paypal, aby bylo možné platbu do rozhraní platební brány odesílat a přijímat výsledky průběhu platby. K odladění byl použit obchodní účet v aplikaci platební brány, pořízený na základě emailové adresy pořízené k vývoji e-shopu použitý i odesílání emailu. Následně v rozhraní pro vývojáře aplikace Paypal byly zřízeny dva sandboxové účty (účty pro bezpečné testování). Tyto dva sandboxové účty byly následně použity při testování funkčnosti zadávané platby, a to díky úvodnímu stavu, kdy oba účty obsahovaly sumu 5000,-, pro tento účel.

4.11.1 Další platební metody

Další platební metody vzhledem k finanční náročnosti nebo scházejících informacích o počátečních nákladech na rozdíl od platební brány Paypal zatím nebyli začleněny. Alternativami by byly například PayU, ThePay, GoPay, ComGate, GP weppay. [13] Platební brána Paypal pouze účtuje poplatek za transakci ve výšce 2,9 % + fixní poplatek 10 Kč.

5 Zhodnocení

Výstupem práce je po technické stránce funkční aplikace e-shopu a doporučení pro kroky ohledně digitálního marketingu.

Aplikace umožňuje uživateli zobrazení detailu produktu, přidání produktu do košíku jak v náhledu, tak v detailu, dále navyšování a snižování počtu produktů v přehledu obsahu košíku. Také si může uživatel vytvořit v aplikaci účet, ke kterému se následně při prvním zadání přiřadí dodací údaje a není jich pak potřeba vyplňovat opakovaně při opakovaném nákupu. Při platbě může uživatel zvolit mezi platbou na fakturu a přes platební bránu Paypal. Uživatelské rozhraní také umožňuje přímo z rozhraní e-shopu odeslat provozovateli emailovou zprávu s dotazem.

Administrátorovi umožňuje aplikace vytváření kategorií produktů, dále konkrétní produkty, u kterých je možno zadat i kolekci fotek produktu, která se pak zobrazuje v detailu produktu. Podobně jako uživatel může i administrátor posílat přímo z rozhraní aplikace emaily, s tím rozdílem, že administrátor má navíc možnost volby cílové emailové adresy. Také administrátorské rozhraní aplikace umožňuje správu objednávek a stažení faktury objednávky z důvodu možnosti jejího tisku k odeslání se zásilkou.

Aplikace pak tu samou emailovou adresu, ze které jsou posílány zprávy z rozhraní aplikace, používá k odeslání zprávy s potvrzením objednávky obsahující i fakturu kterou aplikace generuje.

V rámci vývoje aplikace je ještě potřeba zpracování designové stránky. Tato část aplikace závisí na návrhu samotné majitelky značky a spolupráce s webovým designerem. Toto je samozřejmě v záměru jako další krok na vývoji aplikace e-shopu a pro mne jako pro vývojáře to přesahuje rozsah této práce.

Závěr

Výstupem práce jsou doporučení pro vlastníka značky Išku Fišárkovou, věnovat se budování profilu firmy v digitálním prostředí, zejména pak na sociálních sítích a vytvoření nového prodejního a distribučního kanálu e-shopu. Tento cíl bude průběžně naplňován samotným vlastníkem značky, primárně na sociální síti Facebook. Zároveň bude nejspíš docházet k propagaci na síti Instagram, a to dcerou Išky Fišárkové Adélou Pokornou, která na tvorbě pod značkou MAXFRED začíná spolupracovat. Rozšíření distribučních kanálů o e-shop, který je v rámci této práce po technické stránce aplikace vyřešen, je obsahem digitální přílohy této práce. Po estetické stránce neboli designu e-shopu na něj budou kladeny požadavky odpovídající designu značky MAXFRED, který bude vyvíjen ve spolupráci s web designerem a vlastníkem značky. Mimo to jsou některé údaje například údaje na generované faktuře pouze testovací data pro ověření funkčnosti, ne však data ve finálním stavu. Platební brána Paypal je zatím připojena pouze k sandboxovému účtu použitého k zajištění technické funkčnosti tohoto druhu platby. Také obchodní podmínky budou jako zobrazovaný text doplněny až po jejich vytvoření vlastníkem, povinnost souhlas udělit je již do aplikace začleněna, ale jejich znění chybí. Použití vzorové aplikace z ABC tutorial se projevilo jako funkční řešení, které zjednodušilo návrh základní struktury e-shopu. Zároveň poskytlo dobrý přehled o struktuře zdrojového kódu a v návaznosti na tom jednodušší úpravu pro vlastní specifiky. Potom, co budou tyto kroky doplněny, bude možné e-shop nasadit. Pro e-shop již byla pořízena doména druhého řádu s názvem maxfred.cz.

Seznam zdrojů

- [1] OUDOVÁ, Alena. *Logistika: základy logistiky*. Aktualizované 2. vydání. Prostějov: Computer Media, 2016. ISBN 978-80-7402-238-8.
- [2] GROS, Ivan a kol. *Velká kniha logistiky*. Praha: Vysoká škola chemicko-technologická v Praze, 2016. ISBN 978-80-7080-952-5.
- [3] SKLENÁŘ, Jan. Český rozhlas: Dvojka. Se svým šatníkem si hrajte a hlavně kombinujte, radí módní návrhářka Iška Fišárková: Host do domu [online]. Praha: Český rozhlas, 2010[2021-03-31]. Dostupné také z: <https://dvojka.rozhlas.cz/se-svym-satnikem-si-hrajte-a-hlavne-kombinujte-radi-modni-designerka-iska-7506488>
- [4] REJHONOVÁ, Magdalena. Salon. Nejen pánská móda Išky Fišárkové [online]. 2008 [2021-03-31]. Dostupné také z: <https://www.salon.cz/damska-moda/iska-fisarkova-nove-i-pro-zeny.html>
- [5] VÁVRA, Pavel. Analýza vnitřního a vnějšího prostředí organizace a její využití ve strategickém plánování [online]. Praha: Vysoká škola regionálního rozvoje a bankovní institut – AMBIS, 2020[2021-03-21]. Dostupné také z: https://is.ambis.cz/th/qn54j/bakalarska_prace_Pavel_Vavra.pdf
- [6] ZÁVODNÝ, POSPÍŠIL Jan. Digitální Marketing [online]. Olomouc: Moravská vysoká škola Olomouc, o.p.s., 2018[2021-03-24]. Dostupné také z: <https://www.mvso.cz/files/digitalni-marketing.pdf>
- [7] DOČEKAL, Daniel, Lupa.cz: Server o českém internetu. Sociální sítě si podmanily svět (fakta a čísla hlavně o Evropě) [online]. Praha: Internet Info, s.r.o., 2011[2021-03-31]. Dostupné také z: <https://i.iinfo.cz/images/330/svetova-mapa-socialnich-siti-1.png>. ISSN 1213-0702
- [8] UNGR, Pavel. Pavel Ungr, SEO konzultant: Stopařův průvodce po světě SEO a online marketingu. Nepropadejte panice. Co je to SEO – optimalizace pro vyhledávače? (definice): Optimalizace pro vyhledávače [online]. Praha, 2014 [2021-03-25]. Dostupné také z: <https://www.pavelungr.cz/definice-co-je-seo/>
- [9] TIŠNOVSKÝ, Pavel. Mojefedora.cz. Použití assembleru v Linuxu [online]. Mojefedora.cz 2016[2021-03-31]. Dostupné také z: <https://mojefedora.cz/pouziti-assembleru-v-linuxu/>
- [10] JILEČEK, Jan, Create IT: Cleverance. 5 programovacích jazyků pro začátečníky [online]. Praha: Cleverance Enterprise Solutions, s.r.o., 2019 [2021-03-31]. Dostupné také z: <https://www.create-it.cz/Blog/Stranky/5-jazyku-pro-2020.aspx>
- [11] TUTORIALSTEACHER: Linq Tutorials. Linq Query Syntax [online]. TutorialTeacher.com 2020[2021-03-31]. Dostupné také z: <https://www.tutorialsteacher.com/Content/images/linq/linq-query-syntax.png>

- [12] TUTORIALSTEACHER: Linq Tutorials. Linq Method Syntax [online].
TutorialsTeacher.com 2020[2021-03-31]. Dostupné také z:
<https://www.tutorialsteacher.com/Content/images/linq/linq-method-syntax.png>
- [13] LIŠKA, Lukáš. Návrh a implementace e-shopu pomocí frameworku ASP.NET
[online]. Pardubice: Univerzita Pardubice, 2016 [2021-03-31]. Dostupné také z:
https://dk.upce.cz/bitstream/handle/10195/64935/LiskaL_NavrhAImplementace_RD_2016.pdf?sequence=3&isAllowed=y

Seznam grafických objektů

Obr. 1.1	Struktura distribučního systému	11
Obr. 1.2	Iška Fišárková.....	13
Tab. 1.1	Rozvržení SWOT analýzy.....	16
Obr. 2.1	Mapa sociální sítě ve světě.....	22
Obr. 2.2	Metody a cíle SEO.....	23
Obr. 2.3	Google SEO a SEM.....	24
Obr. 3.1	Nižší programovací jazyk (Assembler).....	27
Obr. 3.2	Vyšší programovací jazyk (Phyton).....	28
Obr. 4.1	Schéma struktury správcovského rozhraní.....	32
Obr. 4.1	Schéma struktury správcovského rozhraní.....	33
Obr. 4.3	LINQ query.....	36
Obr. 4.4	LINQ lambda query	36
Obr. 4.5	Schéma výpočtu datumů expedice.....	38
Obr. 4.3	Vygenerovaná faktura.....	40

Seznam zkratek

ASP – Active Server Pages

B2B – Bussines to bussiness

CLI – Common Language Infrastructure

CSS – Cascading Style Sheets

DCL – Data Control Language

DDL – Data Definition Language

DML – Data Manipulation Language

FSF – Free Software Foundation

GNU – GNU's Not Unix!

HTML – HyperText Markup Language (značkovací jazyk pro tvorbu webových stránek)

IBM – International Business Machines Corporation

IDE – Integrated development environment

IFmode – Iška Fišárková móda

JS – JavaScript

KABO – Obchodní dny obuvi a koženého zboží

Linq – Language-Integrated Query

MS SQL server – Microsoft SQL server

MVC – Model, View, Controller architektura

ORM – object-relation mapping

OSI – Open Source Initiative

OSS – Open source software

PC – Personal computer (osobní počítač)

PDA – Personal digital assistant

PDF – Portable Document Format

PERL – Practical Extraction and Report Language

PESTLE analýza – Political, Economic, Social, Technological, Legal, Environmental (Politická, Ekonomická, Sociologická, Technologická, Legislativní, Ekologická) analýza

PHP – PHP: Hypertext Preprocessor

PR – public realtions (vztahy s veřejností)

SEM – Search Engine Marketing

SEO – Search Engine optimization

SEQUEL – Structured English Query Language

SOFT analýza – Satisfactory, Oportunity, Fault, Threat analýza

SQL – Structured Query Language

STYL – Obchodní dny módy

SWOT analýza – Strenghts, Weakness, Oportunities, Threats analýza

TCC – Transaction Control Comands

UI – User interface

URL – Uniform Resource Locator (webová adresa)

VLC – VideoLAN Client

Seznam příloh

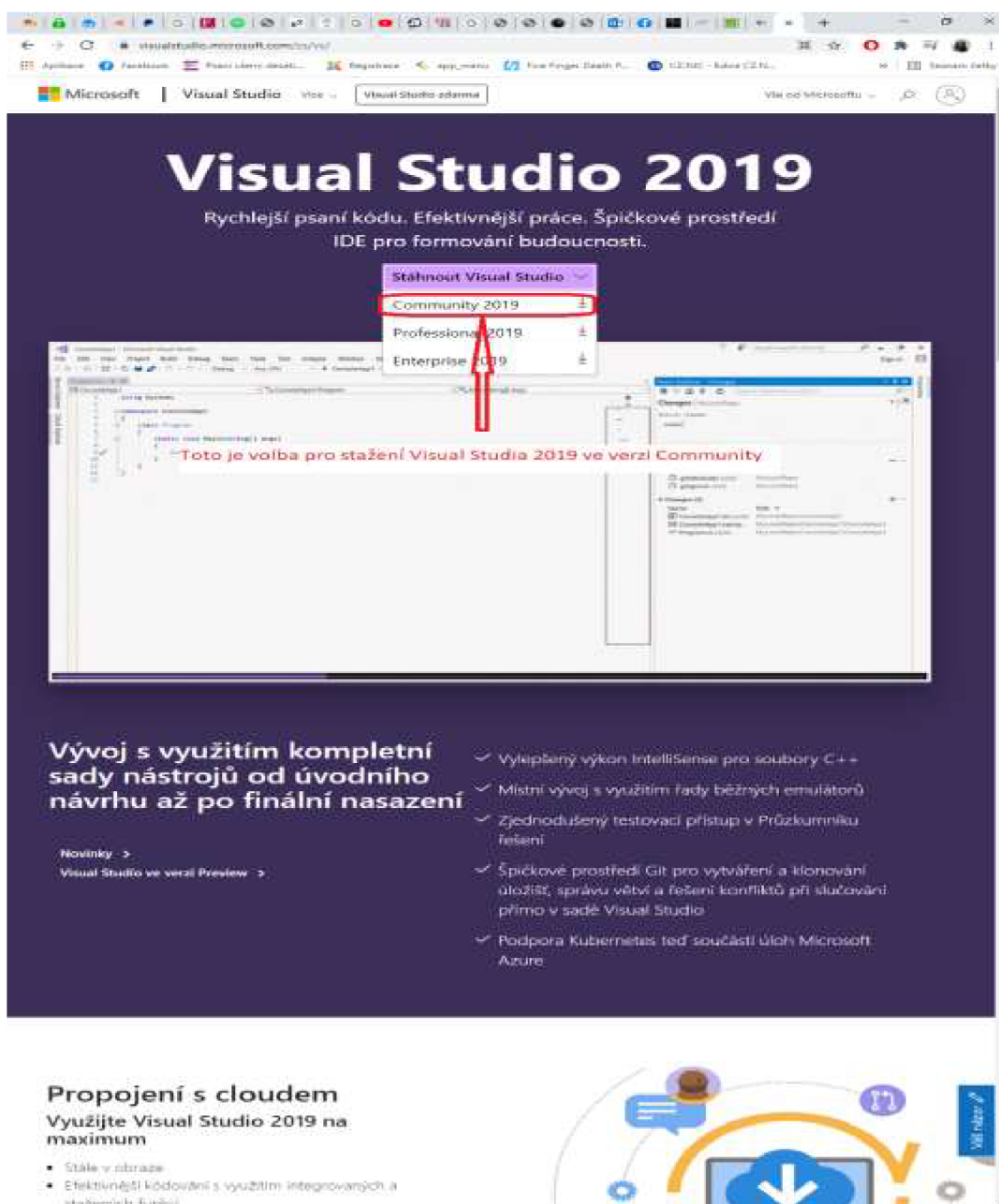
Příloha A – Postup pro zpuštění aplikace e-shopu

Příloha B – program aplikace e-shopu (příloha je vzhledem velikosti souboru k dispozici na Katedře bakalářského studia v digitální formě, v tištěné BP je přiložené CD)

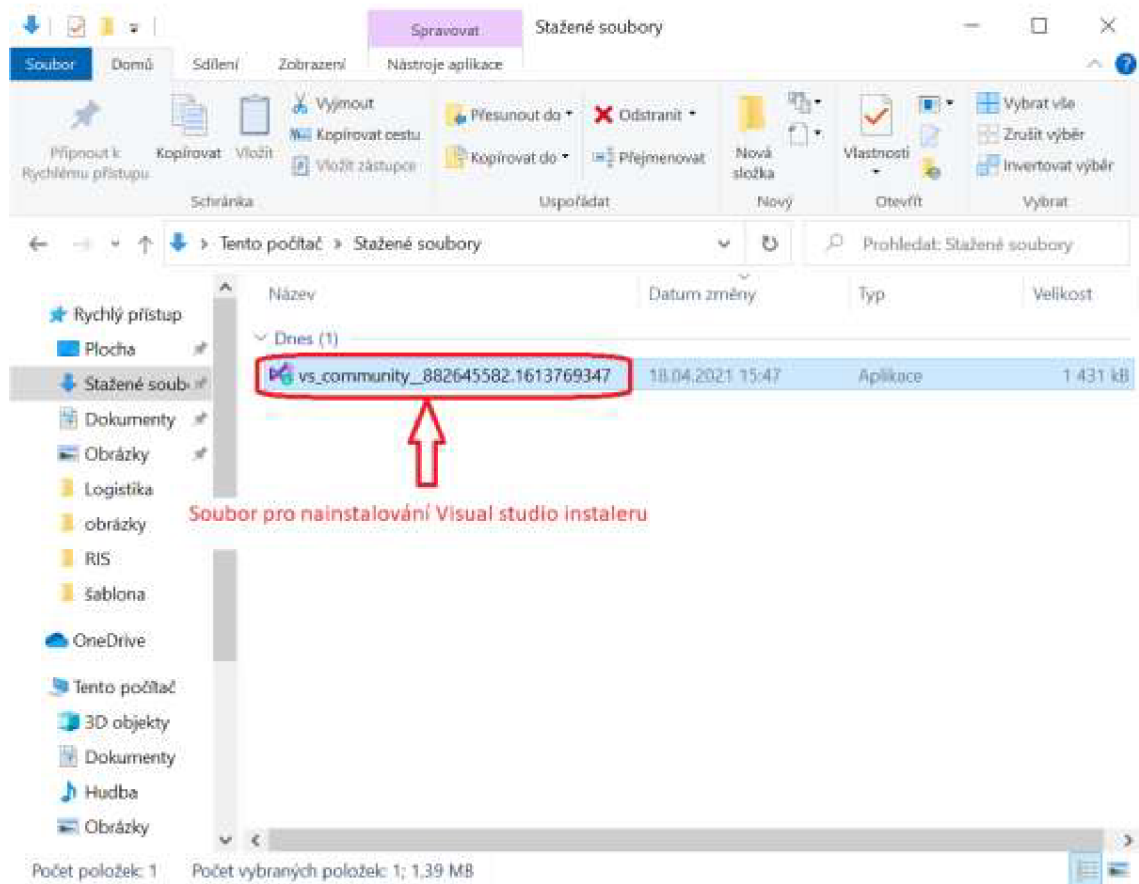
Postup pro spuštění aplikace e-shopu

Pro spuštění nezkompilovaného kódu aplikace e-shopu je nejdříve nutné do operačního systému nainstalovat vývojové prostředí Visual Studio 2019. Verze Visual Studia 2019 – Community je poskytována společností Microsoft bezplatně, a lze ji získat na oficiálních stránkách této společnosti konkrétně na url: <https://visualstudio.microsoft.com/cs/vs/>.

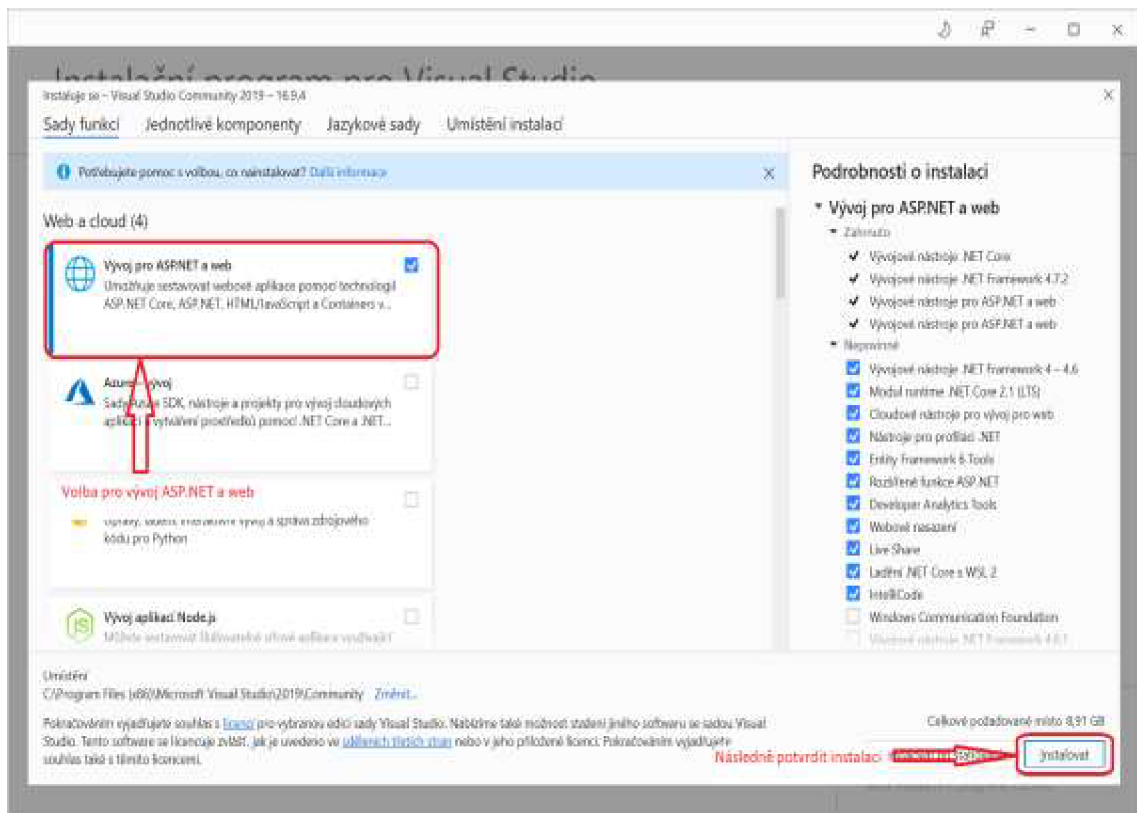
Ostatní verze Visual Studia 2019 jsou zpoplatněné a jejich funkce pro běh aplikace nejsou nezbytné.



Dalším krokem je nainstalování staženého instalačního souboru s názvem vs_community_... .exe

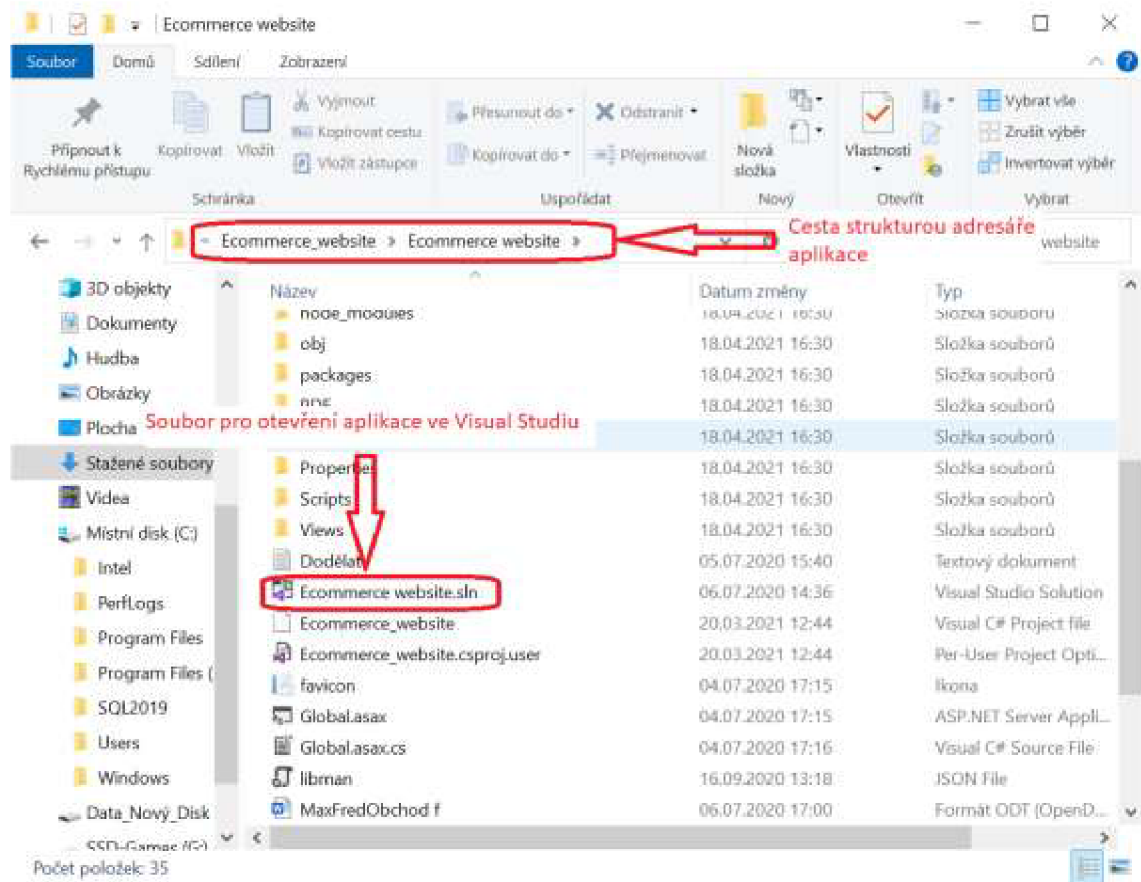


Po nainstalování Visual Studio instaleru se tento instaler otevře a vyzve uživatele k volbě druhu aplikací, pro které má být Visual Studio používáno. Zde je třeba zvolit Vývoj pro ASP.NET a web a poté potvrdit instalaci.

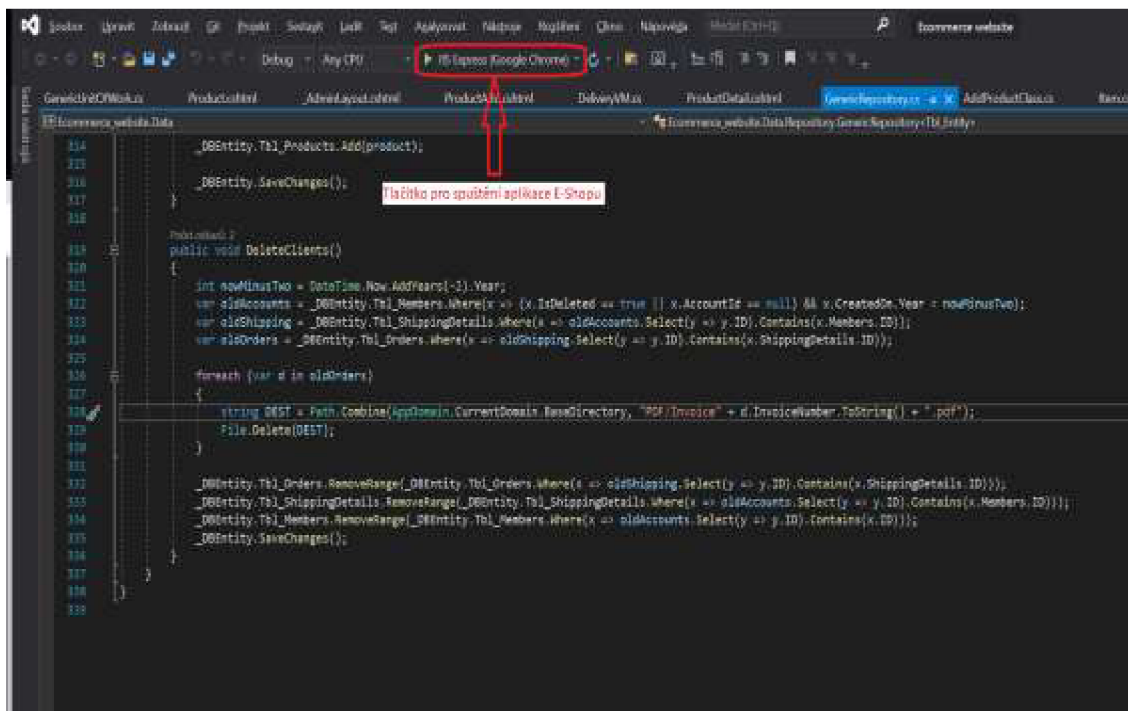


Po dobu instalace je nezbytné, aby systém zůstal připojený k internetu.

Po nainstalování Visual Studia lze instalaci zavřít a přejít do složky aplikace stažené z příloženého CD do libovolného adresáře v systému. V složce řešení aplikace s názvem Ecommerce_website je třeba přejít do projektu se stejným názvem a v něm dát otevřít soubor s názvem Ecommerce_website.sln, tím se spustí vývojové studio a v něm se otevře zdrojový kód aplikace



Po spuštění aplikace ve Visual Studiu již stačí pouze spuštění pomocí zmáčknutí klávesy F5 nebo tlačítka v záhlaví Visual Studia s názvem IIS EXPRESS (+název defaultního prohlížeče).



Pro spuštění administrátorské části aplikace eshopu je třeba se přihlásit jako uživatel s názvem: maxmilianlevy@seznam.cz a použít heslo: HesloEshop@123.

Autor	Maxmilian Fišárek
Název BP	Informační podpora prodejního kanálu malé firmy
Studijní obor	IPL
Rok obhajoby BP	2021
Počet stran	35
Počet příloh	2
Vedoucí BP	doc. Dr. Ing. Oldřich Kodým
Anotace	Tato práce pojednává o vypracování marketingové SWOT analýzy malé firmy a následné zpracování výsledných doporučení pro posílení pozice značky na trhu. Nastiňuje problematiku distribuce, v souvislosti s požadavky vlastníka, probírá téma marketingu v online prostředí. Analyzuje možné způsoby pro vývoj softwaru, konkrétně aplikace e-shopu. Nakonec popisuje způsob řešení tohoto nového distribučního kanálu.
Klíčová slova	Distribuce, SWOT analýza, Digitální marketing, Open Source, E-shop s výrobou na poptávku
Místo uložení	ITC (knihovna) Vysoké školy logistiky v Přerově
Signatura	