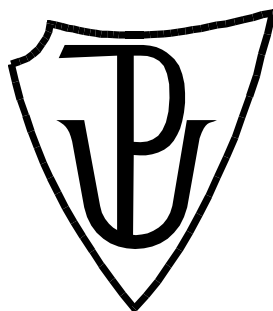


UNIVERZITA PALACKÉHO V OLOMOUCI

Přírodovědecká fakulta

Katedra biochemie



Software pro zpracování a analýzu gelových skenů

BAKALÁŘSKÁ PRÁCE

Autor:	Jaroslav Evják
Studijní program:	B1406 Biochemie
Studijní obor:	Bioinformatika
Forma studia:	Prezenční
Vedoucí práce:	Mgr. Martin Raus, Ph.D
Rok:	2019

Prohlašuji, že jsem bakalářskou práci vypracoval samostatně s vyznačením všech použitých pramenů a spoluautorství. Souhlasím se zveřejněním bakalářské práce podle zákona č. 111/1998 Sb., o vysokých školách, ve znění pozdějších předpisů. Byl jsem seznámen/a s tím, že se na moji práci vztahují práva a povinnosti vyplývající ze zákona č. 121/2000 Sb., autorský zákon, ve znění pozdějších předpisů.

V Olomouci dne

.....

Poděkování

Rád bych poděkoval vedoucímu mé bakalářské práce Mgr. Martinu Rausovi, Ph.D., za odborné vedení, trpělivost, vstřícnost a časovou flexibilitu při konzultacích. Dále Bc. Danielu Fialovi za konzultaci kódu.

Bibliografická identifikace

Jméno a příjmení autora	Jaroslav Evják
Název práce	Software pro zpracování a analýzu gelových skenů
Typ práce	Bakalářská
Pracoviště	Katedra biochemie
Vedoucí práce	Mgr. Martin Raus, Ph.D.
Rok obhajoby práce	2019

Abstrakt

V dnešní době existuje mnoho programů, které se zabývají analýzou vzorků gelové elektroforézy. Většinou se jedná o desktopové aplikace, které dokáží analyzovat jeden gel, jejímž výstupem jsou tabulková data. Získaná data z analýzy mohou sloužit pro případný další výzkum. Práce popisuje princip gelové elektroforézy, typy gelů, zpracování a chyby obrazu, v neposlední řadě existenční řešení. Cílem této bakalářské práce je vytvoření webové aplikace, která umožní automatickou analýzu skenu gelu s následným uložením dat do databáze. Výstupem bude graf, který porovná data zvolených stripů uživatelem.

Klíčová slova	Gelová elektroforéza, zpracování elektroforeogramu, elektroforéza, smilling efekt, oblast píku, standard, detekce píků, UN-SCAN-IT, GelViewer, SoftGel
Počet stran	45
Počet příloh	1 CD
Jazyk	Český

Bibliographical Identification

Author's name	Jaroslav Evják
Title	Software for processing and analysis of gel scans
Type of thesis	Bachelor
Department	Department of Biochemistry
Supervisor	Mgr. Martin Raus, Ph.D.
The year of presentation	2019

Abstract

Nowadays there is many programs that deal with gel electrophoresis sample analysis. Mostly is desktop applications that can analyze single gel whose outputs are spreadsheet data. The data obtained from the analysis can be used for further research. In work with a focus on the principles of gel electrophoresis, mistakes and processing of image, not least existential solutions. The aim of this thesis is to create a web application that would allow automatic analysis of information and subsequent stored data into the database. The output will be a graph that compares the data on strips user.

Keywords	Gel electrophoresis, electrophoresis processing, electrophoresis, smilling effect, peak area, standard, peak detection, UN-SCAN-IT, GelViewer, SoftGel.
----------	---

Number of pages	45
Number of appendices	1 CD
Language	Czech

OBSAH

1	ÚVOD	8
2	SOUČASNÝ STAV ŘEŠENÉ PROBLEMATIKY	9
2.1	WEBOVÉ APLIKACE	9
2.2	GELOVÁ ELEKTROFORÉZA	10
2.2.1	PRINCIPY METODY	10
2.2.2	TYPY GELŮ V ELEKTROFORÉZE	12
2.2.2.1	Polyakrylamidový gel	12
2.2.2.2	Agarózový gel	13
2.3	VÝSLEDKY A POČÍTAČOVÉ ZPRACOVÁNÍ	13
2.3.1	ZKRESLENÍ OBRAZU	14
2.3.1.1	Smiling efekt	15
2.3.1.2	Soudkový a poduškový efekt	15
2.4	EXISTUJÍCÍ ŘEŠENÍ	16
2.4.1	UN-SCAN-IT	16
2.4.1.1	Lanes	16
2.4.1.2	Segment	17
2.4.1.3	Klady a zápory softwaru	18
2.4.2	GELANALYZER	18
2.4.2.1	Klady a zápory	19
2.5	REPREZENTACE OBRAZU V POČÍTAČI	20
2.5.1	BAREVNÉ MODELY	21
2.5.2	RGB MODEL	21
3	PRAKTICKÁ ČÁST	24
3.1	POUŽITÉ TECHNOLOGIE	24
3.1.1	CSS	25
3.1.2	PHP	25
3.1.3	JAVASCRIPT	26
3.1.4	JSON	26
3.1.5	REST API	27
3.1.5.1	GET	28
3.1.5.2	POST	29
3.1.5.3	DELETE	29
3.1.5.4	PUT (Update)	29
3.1.6	MYSQL	30
3.2	NÁVRH DATABÁZE SOFTGEL	31
3.3	TVORBA WEBOVÉHO ROZHRAŇÍ	32
3.3.1	SOUBOROVÁ STRUKTURA SOFTWARE	32
4	VÝSLEDKY A DISKUZE	34
4.1	NAHRÁNÍ GELU	34
4.2	DETEKCE PÍKŮ	36
4.2.1	VÝPOČET STANDARŮ	36
4.2.2	DETEKCE OBLASTI PÍKU	37
4.3	GRAFICKÉ POROVNÁNÍ STRIPŮ	41
5	ZÁVĚR	42
6	LITERATURA	43

Cíle práce

Teoretická část

- Vypracování literární rešerše, která shrnuje přehled o aktuálním stavu softwaru/nástrojů pro počítačové zpracování gelových skenů.
- Obecný přehled technického pozadí dané problematiky.

Praktická část

- Vytvoření softwarového nástroje schopného zpracovat gelové skeny z výchozích vstupních formátů (bitmapové obrazy) do datové formy vhodné pro další hromadné zpracování.
- Součástí práce je objasnění principu fungování jednotlivých částí vytvořeného nástroje.

1 ÚVOD

Předmětem této bakalářské práce je tvorba webové aplikace sloužící k analýze a následnému porovnání skenů, které jsou výsledkem gelové elektroforézy. Tím se dostáváme k oblasti informatiky, kterou je zpracování obrazu. Klíčový důvod pro vytvoření této aplikace vznikl z potřeby porovnávání podobnosti mezi různými gely případně jednotlivými spektry. Takové porovnání je bez využití výpočetní techniky nesmírně složité až téměř nemožné. Převod daných skenů do digitalizované podoby a jejich následné využití společně s dnešní výpočetní technikou ovšem tento problém eliminuje a máme tak možnost přesně a rychle určit místa podobnosti vybraných snímků gelů.

V současnosti již lze v oblasti zabývající se touto problematikou nalézt několik softwarových nástrojů, které podobnou práci se skeny umožňují, ovšem žádný z nich neumožňuje přímo porovnávání gelů mezi sebou a jednotlivými spektry. Problémem těchto již existujících nástrojů je dostupnost a omezenost na platformu. Z většiny se jedná o programy koncipované pouze pro Windows a využití na jiných platformách, jako je Linux může být poměrně komplikované. Problematické může být i jejich komerční povaha, kdy pro plné využití je nutné zakoupení licence a následné platby za aktualizace, což znamená nutné vynaložení finančních prostředků.

Tato aplikace těmito problémy zatížená není, jelikož se jedná o webovou aplikaci, tedy provozovanou plně přes webový prohlížeč na uživatelské stanici. Díky tomu ji není problém využívat na různých platformách, současně nevyžaduje žádnou instalaci a jediné nutné předpoklady k provozu jsou dostatečně aktuální prohlížeč a připojení k internetu. Uživatel tak může využít libovolný počítač s připojením k internetu a začít aplikaci užívat.

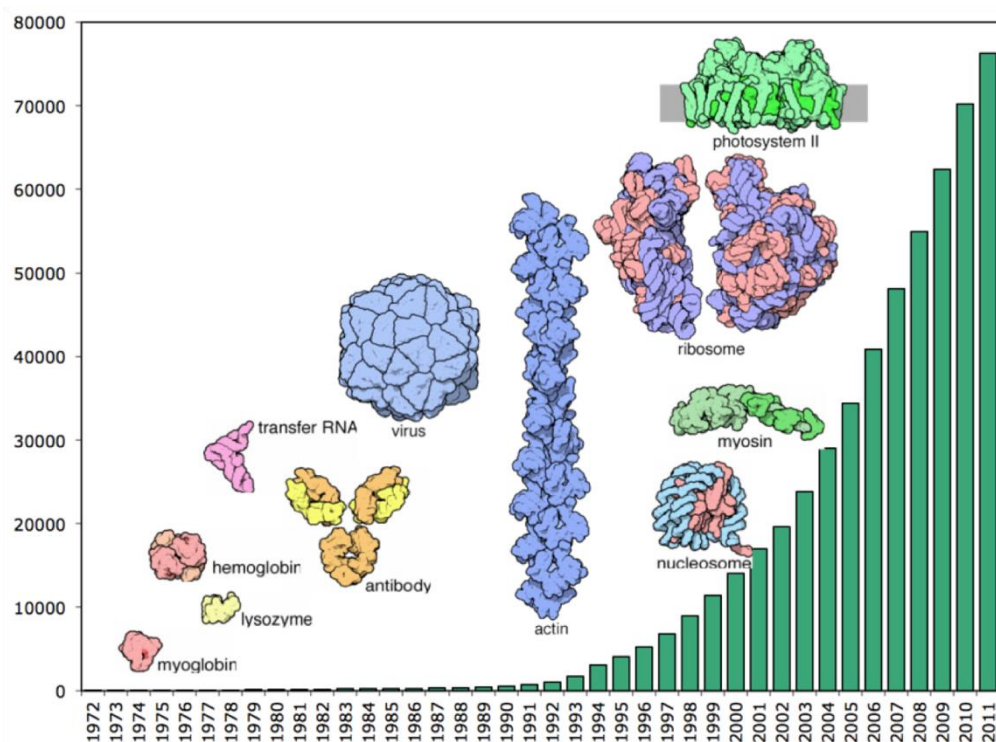
Z funkčního hlediska se jedná o webovou aplikaci založenou na současných standardech s moderním uživatelským rozhraním, která umožňuje shromažďovat skeny gelů a jejich zmíněné porovnávání.

2 SOUČASNÝ STAV ŘEŠENÉ PROBLEMATIKY

2.1 Webové aplikace

V současné době jsou webové aplikace oblíbené vzhledem k jejich snadné dostupnosti. Aplikace je uživatelům dostupná přes webový server skrz počítačovou síť – internet. Možnost využívat aplikace bez nutnosti instalování softwaru na uživatelský počítač je hlavním důvodem k oblíbenosti webových aplikací. V dřívějších dobách byla většina aplikací typu klient-server, což znamenalo, že každá aplikace měla svůj vlastní klientský program, který fungoval jako uživatelské rozhraní a musel být nainstalován na osobním počítači uživatele. Největší rozšíření webových aplikací v odvětví bioinformatiky přišlo s rozvojem nových bioinformatických metod a analýz, které způsobily generování velkého množství dat s následnou potřebou uložit a organizovat data. Díky této skutečnosti vzniklo mnoho webových aplikací, které slouží například jako biologické databáze ve kterých je možno vyhledávat, filtrovat, vkládat či exportovat data pro další výzkum.

Biologické databáze tvoří největší množství webových aplikací v odvětví biologie. Jsou to knihovny biologických dat, které vznikly z vědeckých experimentů, experimentálních technologií nebo počítačových analýz. Většina těchto databází obsahuje data z oblastí proteomiky, genomiky a metabolomiky. Jedná se o data informující o struktuře, lokalizaci a funkcích genů a dalších látek. Pro výzkum jsou nesmírně důležité, protože usnadňují přístup k předchozím výsledkům, které jsou nezbytné pro plánování budoucích výzkumů. Mezi velice známe patří například **PDB** (**Protein Data Bank**), primární databáze 3D struktur makromolekul, která při vzniku (1971) obsahovala pouze sedm struktur. V dnešní době je PDB důležitou bioinformatickou databází, která obsahuje data pro molekulární biologii, medicínu a farmacii. Všechny struktury v databázi jsou volně dostupné a vložit nové struktury do databáze může kdokoliv. Není tedy zaručena bezchybnost dat. Zdrojem dat jsou většinou experimentální měření. Počet struktur v databázi PDB se již od založení každým rokem rychle zvyšuje (Obr.1), (Kouranov *et al.*, 2006).



Obrázek 1: Grafické znázornění nárůstu počtu struktur makromolekul v databázi PDB (převzato ze statistik PDB, 2019).

2.2 Gelová elektroforéza

Pomocí gelové elektroforézy získáváme možnost separovat proteiny nebo molekuly DNA či RNA. Různými způsoby můžeme následně tyto jednotlivé fragmenty dále analyzovat. Elektroforéza se řadí mezi metody užívané pro analytické a preparativní účely. Jedná se o vysoce účinnou metodu při rozdělování makromolekul o různých velikostech a nábojích. Další její využití je například k zjištění charakteru povrchů organismů jako jsou bakterie nebo viry.

Ke konci devatenáctého století bylo objeveno, že v elektrickém poli se pohybují anorganické částice koloidního roztoku. Pár let poté byl dokázán pohyb proteinů ve vodném roztoku na stejném principu. Ve dvacátém století byla dále rozvíjena tato metoda, která na částice v matici aplikovala elektrické pole a tím způsobila jejich separaci (Gaš, 2001).

2.2.1 Principy metody

Elektroforéza je založena na odlišném pohybu nabitých částic v elektrickém poli, díky tomu je možné částice následně separovat. Elektrické pole, které působí mezi dvěma

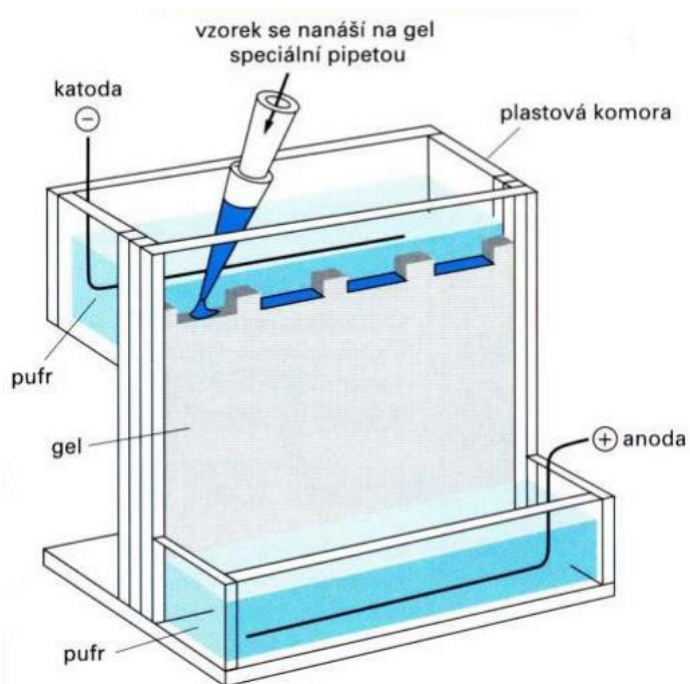
elektrodami vzniklo vložení konstantního stejnosměrného napětí. Jedním z hlavních faktorů ovlivňující elektroforézu je napětí, pokud máme nízké napětí, snižuje to pohyblivost DNA na gelu, optimální hodnota napětí je 5 V/cm. K dalším faktorům patří například teplota (Klouda, 1996).

Jako nosič vzorku se používá nejčastěji agarózový nebo polyakrylamidový gel, na který se vloží daný vzorek. Dalším krokem je vložení vzorku do pufru, který má funkci udržení stálého pH v průběhu celé gelové elektroforézy. Nosiče představují přechod mezi pevným a kapalným stavem, jejich struktura je tvořena z velkého množství vodou.

Pohyb nabitých částic v elektrickém poli závisí na mnoha faktorech, například na molekulové hmotnosti, náboji molekuly, intenzitě elektrického pole a také na pórovitosti gelu. Pórovitost je vlastnost gelu, kdy gel obsahuje drobné a rovnoměrné dutiny skrz, které procházejí částice molekuly dle jejich velikosti. Velikost proteinů či nukleonových kyselin odpovídá tloušťce pórů. K separaci částic dochází v průběhu gelové elektroforézy. Pokud na začátku všechny částice vychází ze stejné pozice jejich oddělení závisí na elektroforetické pohyblivosti (Kealey, 2001).

$$\mu_e = \frac{q}{6\pi\eta r}$$

Kde q představuje náboj částice, η viskozitu elektrolytu a r poloměr částice. Ze vztahu vyplývá, že menší částice s větším nábojem se pohybuje nejlépe. Částice s větší pohyblivostí se oddělí do popředí od částic s menší pohyblivostí. Tato metoda se často využívá pro separaci biologických vzorků, nukleonových kyselin či bílkovin (Obr 5.), (Pouchlý, 2008).



Obrázek 5: Gelová elektroforéza (převzato z ach.upol.cz 2019)

2.2.2 Typy gelů v elektroforéze

Mezi nejvíce používané nosiče při gelové elektroforéze patří agarózový gel nebo polyakrylamidový gel. Nosič je impregnován elektrolytem, díky kterému jsou separované částice na něj upevněné. Pokud elektroforéza probíhá ve volném roztoku, separaci látek může znesnadnit difúze, která má snahu stabilizovat koncentraci (Klouda, 1996).

2.2.2.1 Polyakrylamidový gel

Mezi velký posun v separačních metodách patří používání gelu, který se skládá ze sítě se základem polyakrylamidu. Skrze tento gel se zavedla metoda PAGE (Polyakrylamidová gelová elektroforéze). Velkou předností této metody je, že má vysoce rozpoznávací schopnost, díky které je schopna porovnávat a separovat velice malé molekuly o velikosti bp. Mezi další výhody lze zahrnout, že nemění chemické složení DNA oproti gelům složených z agarózy. PAGE je velice oblíbená mezi biology a biochemiky, protože DNA oddělená z gelu je ve velice dobrém stavu a okamžitě připravená k dalším analýzám.

K nevýhodě oproti agarózovému gelu patří zejména příprava a možnost zranění, protože polyakrylamidový gel se skládá z toxických látek. Při přípravě

polyakrylamidového gelu je potřeba používat ochranné rukavice a ostatní doporučené oblečení (Berčík, 1997).

2.2.2.2 Agarózový gel

Agarózu představuje lineární sacharidový polymer D-galaktosidázy a 3,6-L-galaktopyranozy. Obsah agarózy v gelu se pohybuje okolo 0,5% až 4%, od toho se odvíjí separační schopnost gelu, která je vyšší na základě obsahu polysacharidu. Molekuly o jiné velikosti se pohybují na agarózovém gelu s jinou koncentrací odlišnou rychlostí. Díky tomu s rostoucím obsahem agarózy také klesá rychlost elektroforézy a příprava gelu je také obtížnější. Separace je sice pomalejší, ale navzdory tomu můžeme očekávat lepší separaci fragmentů.

K obrovským výhodám patří bezesporu snadná příprava, kdy smícháme agarózu v práškové formě s pufrem a vzniklý roztok uvedeme k varu. Poté po snížení teploty nalijeme do příslušné formy. Práce s gelem je oproti polyakrylamidovému gelu bezpečná.

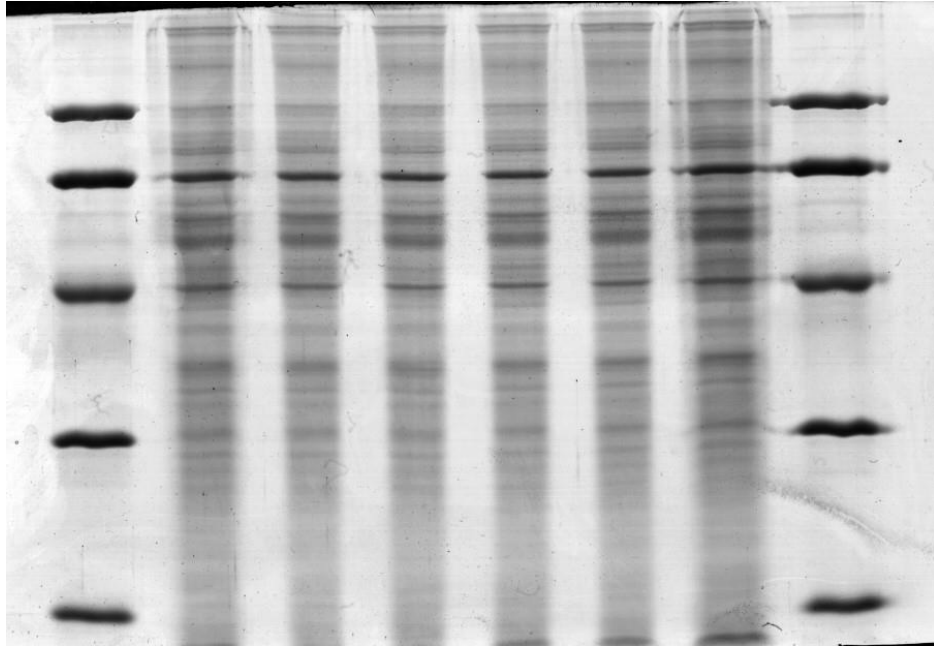
Velkou nevýhodou je zejména přírodní původ agarózy, kdy jednotlivé produkty se mohou od sebe lišit i když jsou od stejného výrobce. To může zapříčinit odlišné výsledky získané z tohoto gelu na základě znečištění ionty kovů nebo sacharidy. To může vést k inhibování dalších reakcí (Jeppson *et al* 1979).

2.3 Výsledky a počítačové zpracování

V konečné fázi, kdy gel obsahuje už separované molekuly ho obarvíme UV činidlem. Při analýze DNA na agarózovém gelu se využívá především ethidium bromid. Ethidium bromid, který je navázaný na DNA při ozáření ultrafialovým zářením, emituje oranžové světlo, díky čemuž je poté možné vidět výsledek. U analýzy proteinů se využívá barvivo zvané **Coomassie** nebo **Brilliant Blue**, což je modré barvivo používané především na polyakrylamidovém gelu v přítomnosti dodecylsírany sodného.

Dalším krokem je pořízení digitálního obrazu pomocí digitálním fotoaparátu nebo skeneru. Existuje mnoho doporučených postupů, které je nutno dodržovat, aby se pořízené snímky mohly lépe analyzovat. Nedodržování může vést k získání různých obrazů z téhož gelu, což vede k zavádějícím výsledkům softwarové analýzy.

Vzhledem k monitoru musí být pohyb částic v gelu vertikálně orientován, aby se předešlo následnému otáčení obrazu. Výsledek, pokud je možno, ukládáme v bezztrátovém formátu PNG (Obr 6.), (Storchová, 1998).



Obrázek 6: Výstupní obraz elektroforézy

Volba ztrátových formátů vede k nepřesným výsledkům. Výsledkem je černobílý obraz, kde jsou znázorněny shluky separovaných molekul. Na levé a pravé straně obrazu se nachází vodorovné čáry představující standardy (velikost markeru), pomocí nichž lze odhadnout velikost fragmentů vzorku. Mezi standardy se nachází šest jamek, které obsahují zkoumané vzorky DNA. Vzorky zde představují svislé čáry, které vznikly putováním různých částic po drahách. Jejich poloha souvisí s nábojem molekuly, její velikostí a tvarem. Počet jamek a standardů není vždy pevně daný a může se lišit na základě potřebné analýzy. Při softwarovém zpracování budeme pracovat se vzniklými svislými čárami vzorku a vodorovnými čárami standardu, kde vše ostatní se bere jako vzniklý šum, který je potřeba odfiltrovat pomocí softwaru.

2.3.1 Zkreslení obrazu

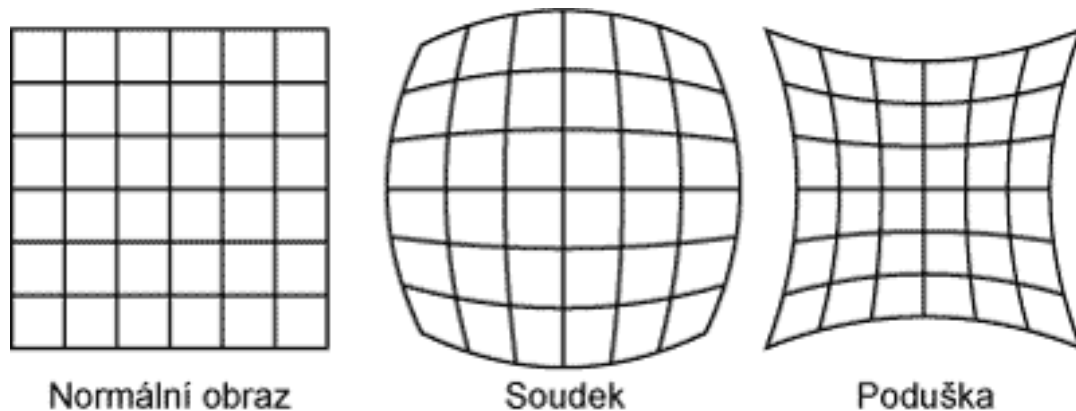
Mezi nejčastější důsledky u fotografování patří nedokonalostí optiky objektivu. U gelových skenerů zase z mechanických důvodů například křivé položení na skener nebo už v samotném gelu. Tyto důsledky se projeví následně na zhoršení kvality obrazu a jeho následném počítačovém zpracování.

2.3.1.1 Smiling efekt

Při elektroforéze aparaturou putuje stejnosměrný proud, který má za následek tvoření tepla, což způsobuje ohřívání daného gelu a může poškodit zkoumané bílkoviny nebo zanechat morfologické stopy, které se projeví ve výsledném obraze. Protože míra teploty je přímo úměrná proudu, který putuje aparaturou, je potřeba nastavit správné hodnoty výstupního napětí el. zdroje. Pokud bude vysoká teplota, separace látek bude trvat déle a dojde k zhoršení rozlišení částic, proto je nutné soustavu dobře chladit. Není-li teplo dobře odváděno, může vznikat smiling efekt, který tvoří nerovnoměrné rozložení svislých čar po stranách obrazu (Jedra *et al*, 2019).

2.3.1.2 Soudkový a poduškový efekt

Soudkové zkreslení často vzniká díky širokoúhlým objektivům, projevuje se zakřivením linií směrem ze středu ven od snímku. Opakem soudkového zkreslení je poduškové zkreslení, které vzniká převážně díky teleobjektivům. Zde jsou linie směrem do středu obrazu (Obr 7.), (Digimanie.cz 2006)



Obrázek 7: Rozdíl mezi normálním obrazem, obrazem se soudkovým a poduškovým efektem (převzato z Fotoroman, 2019)

2.4 Existující řešení

V současnosti je mnoho dostupných softwarů pro analýzu výsledku z gelové elektroforézy. Problém jsou spíše možnosti daného softwaru, protože většina z nich funguje velmi obecně a není specifikovaná na danou analýzu, která je případně vyžadována. Mezi další problém se řadí stáří softwaru, kdy některé jsou z doby Windows 7, tedy přibližně 10 let staré. Tato skutečnost se projevuje na uživatelském rozhraní, které působí zastaralé. Dále se zde využívají se zde staré technologie a postupy analýzy. Popsané softwary v následujících podkapitolách jsou dodnes podporované, ale v případných aktualizacích jde spíše o opravu dílčích chyb než o přínos nových metod.

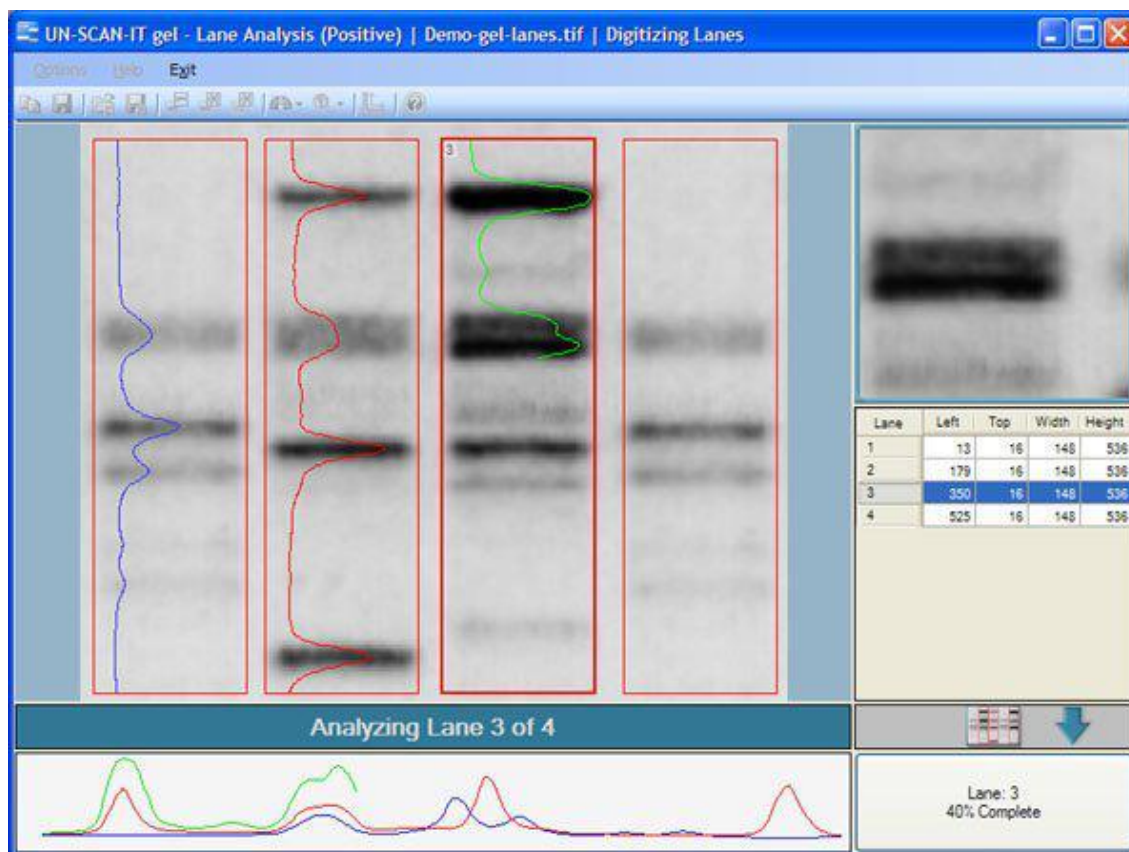
2.4.1 UN-SCAN-IT

Software je vyvíjený od roku 1994 americkou společností Silk Scientific, Inc. Snahou společnosti je nahrazení drahých systémů, a to umožněním analýzy elektroforézních gelů přímo ze standartního skeneru. Od vydání gelového softwaru je ve snaze software neustále aktualizovat a obohacovat o nové uživatelské funkce, které mají za následek, že UN-SCAN-IT je v analýze rychlejší a intuitivnější. V současnosti se software využívá ve více než 80 zemích. Jedná se o desktopový software, který je podporován pro systémy Windows a Macintosh. Poslední verze 7.1 byla vydána v roce 2014 a je nadále společností podporovaná. V plánu je mnoho nových funkcí, mezi které patří například přiblížení okna s gelem, měření vzdáleností a jiné.

Samostatná instalace je pro běžného uživatele jednoduchá, avšak zdarma je dostupná pouze demoverze, která obsahuje pár základních funkcí. Pokud uživatel chce využít UN-SCAN-IT plnohodnotně, je nucen zaplatit každou aktualizaci na vyšší verzi. Software i v demoverzi nabízí pro základní analýzu dostačující funkce (Silkscientific, 2019)

2.4.1.1 Lanes

Jedním ze tří dostupných funkcí v demoverzi je analýza vertikálních čar (dráhy částic). Tato funkce provádí analýzu každého pruhu zvlášť, je zde také možnost volby volitelného počtu pruhů, které chceme zpracovat. Každý takový pruh může obsahovat buď jeden nebo více segmentů (Obr 7.).

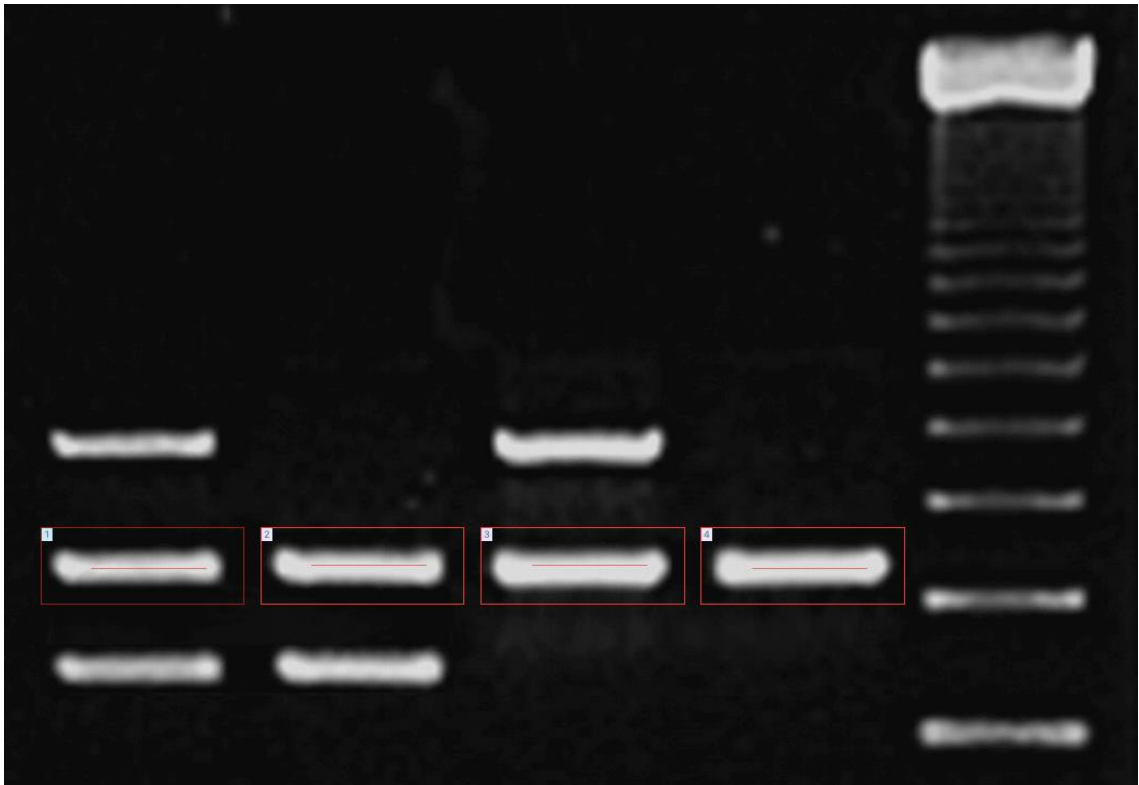


Obrázek 7: Znáornění průběhu funkce Lanes

Pro každý pruh, a segmenty v něm, software spočítá všechny intenzity pixelů a tyto součtové hodnoty pro každý pruh jsou zobrazeny pomocí grafu s osami x a y . Software pak automaticky najde vrcholy v grafu, které odpovídají každému segmentu. Výsledná data je možné exportovat do textového souboru nebo do souboru CSV. Tento soubor je podporován mnoha tabulkovými softwary, jako je například Microsoft Excel. S výslednými daty lze pak nadále provádět potřebnou analýzu.

2.4.1.2 Segment

Pokud je potřeba pracovat pouze s řadou horizontálních segmentů, a ne s celým vertikálním pruhem, je k tomu vhodná právě tato funkce. Tak jako ve funkci Lanes se i zde provede spočítání intenzity pixelů a na základě toho se provede vyhodnocení dat (Obr 8.)



Obrázek 8: Označené segmenty k následné analýze funkcí Segment

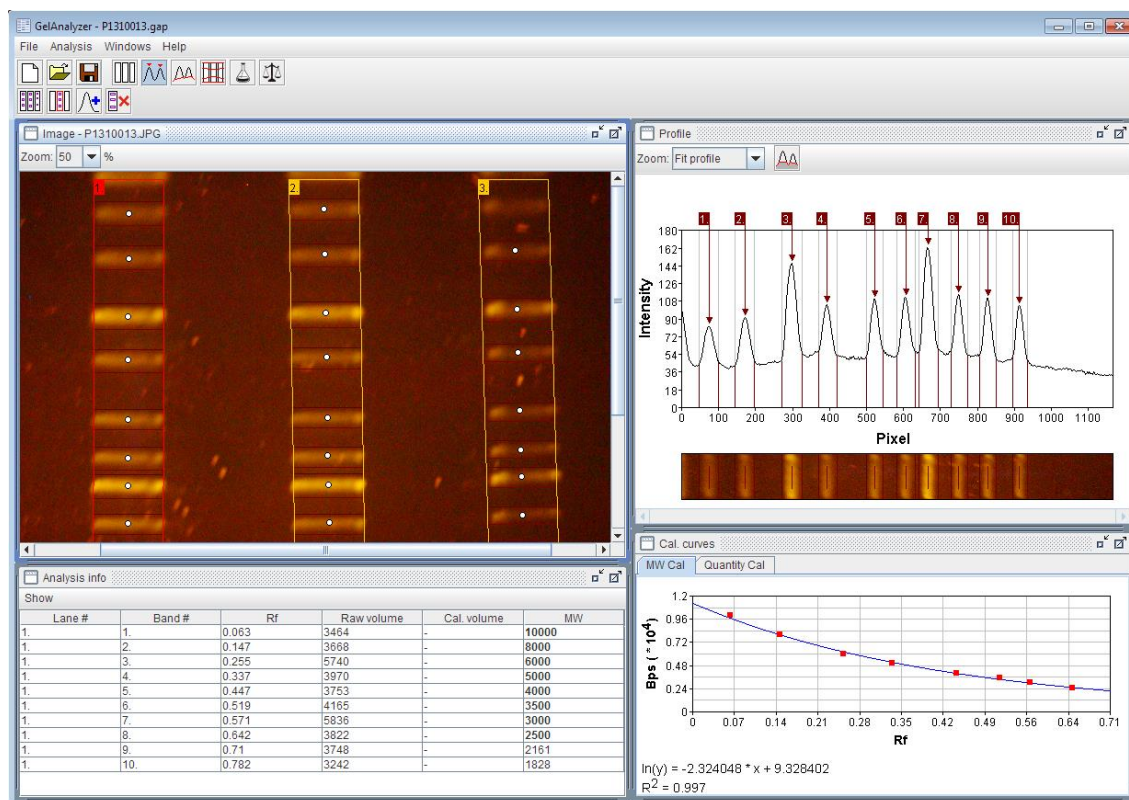
2.4.1.3 Klady a zápory softwaru

K výhodám patří bezesporu stálá podpora, kdy pro software jsou nadále vydávány další aktualizace obsahující nové funkce a nástroje, které usnadňují práci se softwarem. Další výhodou je snadná manipulace se softwarem, kdy všechny funkce a nástroje zvolené uživatelem jsou před prvním použitím uživateli představeny.

Největší nevýhodou softwaru je, že je komerčně založený a zdarma ke stažení je pouze demoverze, která je limitovaná dostupnými funkcemi a nástroji. Každá aktualizace na novou verzi činí 3 500 Kč, zde záleží na uživateli, zda plně využije nové funkce přichozí každou aktualizací nebo ne.

2.4.2 GelAnalyzer

Software vyvinutý firmou LazarSoftware, který vznikl v roce 2010, také nabízí analýzu skenu gelu. Hlavním cílem vývoje byla snaha udržet software co nejjednodušší, což se značně projevilo na uživatelském rozhraní, které vypadá velice jednoduše. Cílovou skupinou uživatelů jsou vzdělávací instituce a malé firmy, na základě toho je zcela zdarma (Obr 9.), (GelAnalyzer, 2019)



Obrázek 9: Detekce pruhů a segmentů v softwaru GelAnalyzer

Rozbor skenu gelu je přímočarý a poskytuje všechny hlavní aspekty hodnocení od výpočtů molekulové hmotnosti až po automatickou detekci pruhů či kalibraci pozadí. Software je vyvíjen v jazyce Java, tudíž běží na jakémkoliv platformě a jedinou nutností je mít nainstalovanou Javu.

Většina softwarů, které umožňují funkce, jako výpočet z pruhu či segmentu, funguje na stejném principu, kdy se pracuje s pixely a jejich barevnou hloubkou. Za zmínku u GelAnalyzera stojí funkce kalibrace obrazu, protože výsledky z gelové elektroforézy bývají poškozené, ať je to už nedodržením postupů při digitálním zpracování či faktory ovlivňující průběh metody. Díky této funkci, lze eliminovat například soudkové nebo poduškové zkreslení a díky tomu dosáhnout mnohem přesnějších výsledkům.

2.4.2.1 Klady a zápory

Ke kladným aspektům jistě patří, že je zcela zdarma a dostupný pro všechny platformy. Uživatelské rozhraní je jednoduché a přehledné.

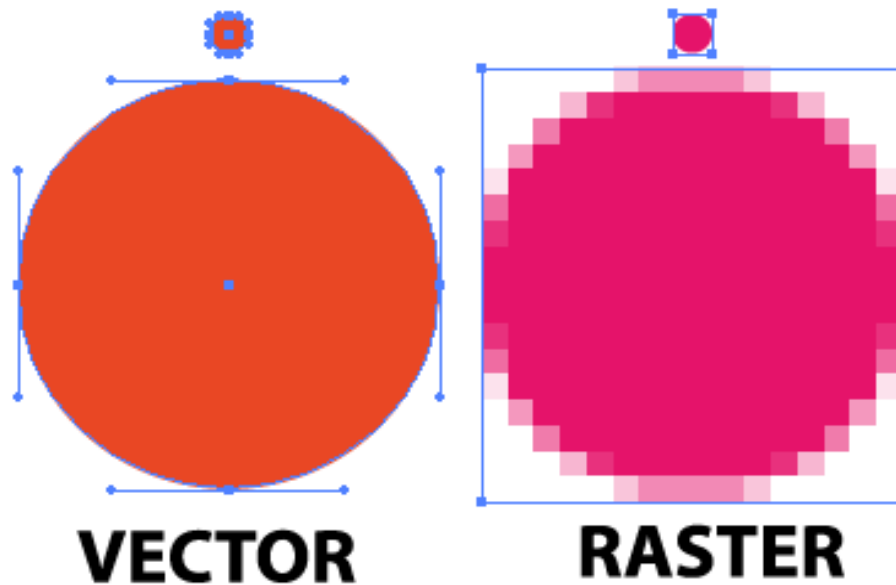
Bohužel software již není podporován, tudíž nelze očekávat aktualizace s novými funkcemi či opravami. K dalším nevýhodám patří nepochybně malý poměr funkcí oproti

UN-SCAN-IT, jelikož jsou dostupné pouze základní funkce pro analýzu. Pro uživatele samostatná instalace a ovládání může být složitější než u předešlého softwaru, protože zde není představeno, jak se taková analýza v softwaru provádí, oproti tomu u UN-SCAN-IT je vše znázorněno a vysvětleno.

2.5 Reprezentace obrazu v počítači

Počítače využívají k reprezentaci obrazu dva typy grafiky – vektorová a rastrová. Rastrová grafika oproti vektorové grafice popisuje obraz pomocí pixelů, které jsou uspořádány do pravoúhlé mřížky, vektorový obrázek je definovaný pomocí základních útvarů jako jsou body, přímky a křivky. Velice rozšířeným vektorovým formátem je například SVG. Předností je jednoduchá manipulace jako zmenšení nebo rotace. Při manipulaci s obrazy nedochází k ztrátě důležitých dat a informací o obrazu. To je značnou výhodou. Vektorová grafika se často používá například pro tvorbu počítačových animací.

Další možností je rastrová reprezentace, kde základní jednotkou je pixel, který popisuje jeden bod obrazu. Na základě toho, že pixely jsou rozmístěny rovnoměrně ve dvourozměrné mřížce, se tato reprezentace nazývá rastrová. Většina obrazových formátů (PNG, BMP, GIF, TIF, JPEG 200) využívá právě tento model. Oproti vektorovému modelu, jakákoliv změna velikosti u rastru znamená, že ztrácíme nebo si dopočítáváme informaci (Obr 2.), (Digimanie.cz, 2019).



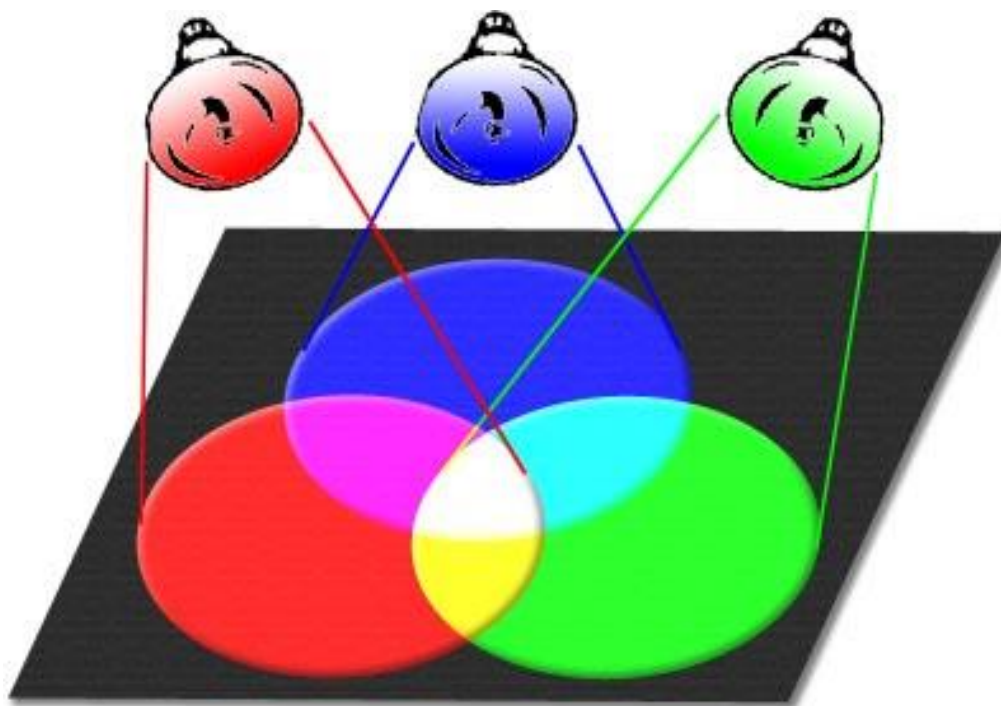
Obrázek 2: Rozdíl mezi vektorovým a rastrovým obrázkem při zvětšení (převzato z Digimanie.cz, 2019)

2.5.1 Barevné modely

Barevné modely popisují, jak pomocí několika základních barevných složek popsat celý barevný prostor. Každou barvu je možné složit pomocí tří základních barevných složek – červená, zelená a modrá. Modely jsou ale omezeny na určitou část barevného prostoru, například zmíněné tři barvy odpovídají modelu RGB, který patří mezi nejvíce používané. Existuje však ještě mnoho jiných barevných modelů, které používají základní barvy jiné nebo barvy jinak skládají. Mezi takové modely patří CMYK (Cyan Magenta Yellow Key), který je založený na subtraktivním míchání barev (mícháním barvy od sebe odečítáme, kdy se s každou přidanou barvou odebrá část původního světla). Využití tohoto modelu je například u tiskáren (Digimanie.cz, 2019).

2.5.2 RGB model

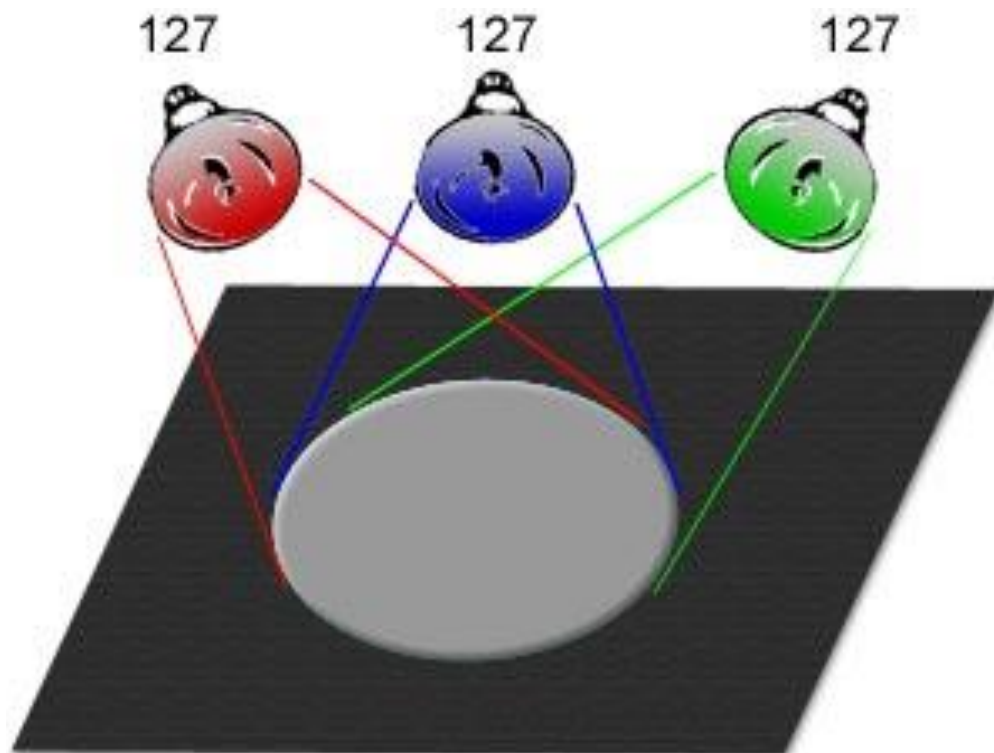
Ve většině případů se pro ukládání a reprezentaci obrázku v počítači používá RGB model. Využívá se masivně od vzniku televize a vychází ze skutečnosti, že zhasnutý monitor je černý a všechny barvy se vytváří mícháním jednotlivých složek – aditivní model. V modelu RGB jsou tři barevné složky červená (red), zelená (green) a modrá (blue), odtud vychází název RGB. Regulací jednotlivých složek svítících na jedno místo je možné dosáhnout různých barev jejich vzájemným mícháním (Obr 3.).



Obrázek 3: Znárodnění modelu RGB, který znázorňuje, jak mícháním tří složek můžeme získat různé barvy. (převzato z Digimanie.cz, 2019).

Pokud svítí všechny tři složky na maximum, místo, kam svítí (tj. pixel) je bílé. Nevýhodou RGB je celkem neintuitivní reprezentace barev a úpravy obrazu uživatelem. Kvůli chybějící jasové složce, který RGB nemá, na rozdíl do ostatních modelů, lze očekávat horší kompresní poměr a není možné popsat určité barvy.

K celému popisu jednoho pixelu je nutný soubor tří čísel, které určují intenzitu složek RGB. Složky ($R=0$, $G=0$, $B=0$) popisují černou barvu, naopak ($R=255$, $G=255$, $B=255$) barvu bílou. Celý sken gel je poté tedy sada trojic čísel, kterých je přesně tolik kolik má sken pixelů. Díky faktu, že výsledek gelového skeneru je ve většině případů šedotónový obrázek, stačí pracovat při porovnávání gelů pouze s jednou složkou ze tří, protože jejich číselné hodnoty jsou stejné (Obr 4.), (Digimanie.cz, 2019).

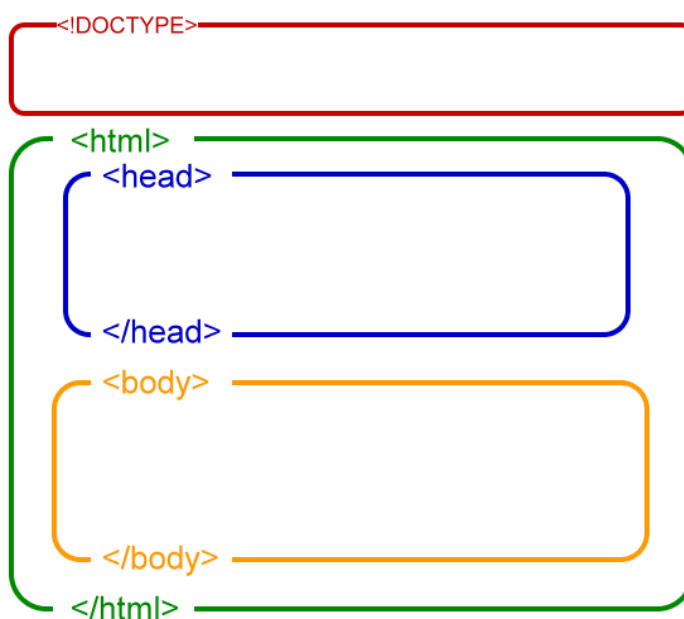


Obrázek 4: Šedotónový obrázek má všechny tři číselné hodnoty stejné, od 0 (černá) až do 255 (bílá). Poloviční hodnota všech světel (127) potom znázorňuje tzv. 50% střední šedou (převzato z Digimanie, 2019).

3 PRAKTICKÁ ČÁST

3.1 Použité technologie

HTML (HyperText Markup Language) je jazyk, který se označuje za značkovací jazyk a je charakterizován pomocí značek (tagů) a jejich vlastností (atributů). Vychází z dřívějšího SGML (Standard Generalized Markup Language) a využívá se již od roku 1990 pro tvorbu webových stránek. V současnosti je takřka samozřejmostí každého webu. HTML je multiplatformní, tím pádem nezáleží, jaký operační systém uživatel používá. Hlavní podmínkou ke správné funkčnosti je mít aktualizovaný webový prohlížeč. Značky definují částem textu určitý význam, pomocí úhlových závorek < a > se uzavírají vlastnosti a názvy značek. Značky bývají párové, kdy většinou druhá značka se označuje jako koncová, akorát má před názvem zpětné lomítko. Například značka <i> změní text mezi značkami na kurzívu, výsledkem <i>Bioinformatika</i> bude *Bioinformatika*. Jazyk slouží k tvorbě struktury webu (Obr 10.), (Pilgrim, 2010).



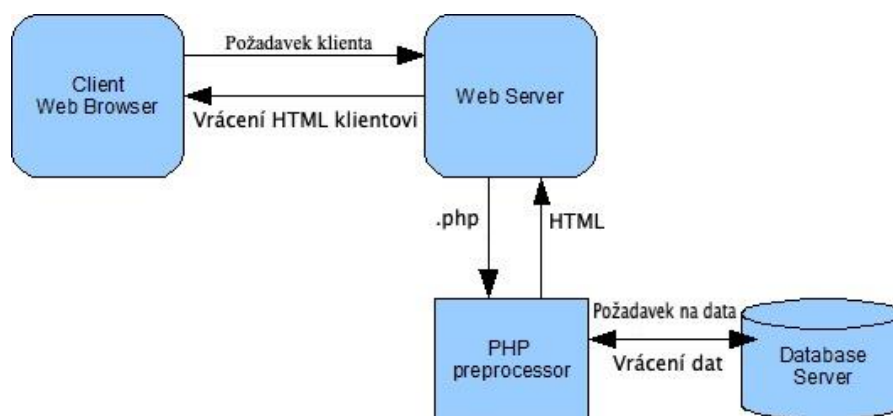
Obrázek 10: Přehled základní struktury webu, značka !DOCTYPE určuje podle jaké definice má být dokument zpracován, <head> definuje hlavičku a <body> obsahuje veškerý zobrazovaný obsah stránky.

3.1.1 CSS

Cascading Style Sheets (zkráceně CSS), nebo-li kaskádové styly, slouží k upravení vizuální podoby stránky. Přiřazují jednotlivým prvkům z HTML dokumentu vzhled či efekty. Například můžeme měnit barvu pozadí, písma, přidat ohraničení nebo různé efekty, které jsou v dnešní době oblíbené. Uživateli přijde HTML dokument, který obsahuje odkaz na kaskádové styly, podle kterých se dokument nastyluje (Sklyr, 2001).

3.1.2 PHP

PHP (zkráceně Personal Home Page), na rozdíl od zmiňovaného CSS a HTML, pracuje na straně serveru. Zpracování požadavku uživatele obstarává server, kdy uživatel pošle požadavek na server a jako odpověď dostane zpracovanou stránku ve formátu HTML. Díky tomu se o PHP mluví jako o hypertextovém preprocesoru. PHP vytváří dynamicky generovaný soubor, který na základě potřebných dat uživatelem vytvoří šablonu a naplní ji daty. Tenhle postup má obrovskou výhodu, kdy nedochází k redundanci kódu, snadno se upravuje a šetří paměť (Obr 11.), (Gutmans *et al.*, 2015).



Obrázek 11: Komunikace mezi klientem, PHP preprocesorem a serverem.

3.1.3 JavaScript

Jedná se o multiplatformní, objektově orientovaný, skriptovací jazyk. Syntaxí se podobá jazykům jako je C nebo Java, ale s jazykem Java nemá nic společné, jedná se pouze o marketingový název. JavaScript je oproti zmíněným jazykům dynamicky typovaný, což znamená, že o typech proměnných se rozhoduje až za chodu programu. Přestože neobsahuje klasický koncept třída-objekt, tahle vlastnost je zde nahrazena prototypováním, což znamená, že je schopen simulovat mnoho třídově založených vlastností. Mezi ně patří dědičnost, která já zde nazývána jako dědičnost prototypová.

Hlavním rozdílem je, že se spouští na straně klienta, na rozdíl například od PHP, díky tomu nezatěžuje server, ale ke zpracování daného kódu dochází na straně klienta. Pomocí něj jsou často ovládány interaktivní prvky (textové pole, tlačítka), animace nebo obrázky. Umožňuje například kontrolovat správnost vyplněných údajů do textové formuláře uživatelem. Je potřeba mít na paměti, že tahle kontrola lze obejít, a proto je potřeba údaje kontrolovat i na straně serveru. K oblíbeným knihovnám, které přidávají či upravují funkce patří jQuery nebo Node.js, který lze využít i na straně serveru (Žára, 2015).

V softwaru se používá knihovna CanvasJS, které slouží pro vykreslování grafů s možnými efekty a filtry. Je použita pod licenci, jenž poskytují pro studenty (CanvasJS, 2019).

3.1.4 JSON

JavaScript Object Notation (zkráceně JSON) je způsob zápisu dat, který je slouží pro přenos dat a je multiplatformní. Data jsou organizovaná do polí a objektů. Vstupem je libovolná struktura dat (číslo, řetězec a jiné), kdy výstupem je vždy řetězec (Obr. 12), (Bassett, 2015).

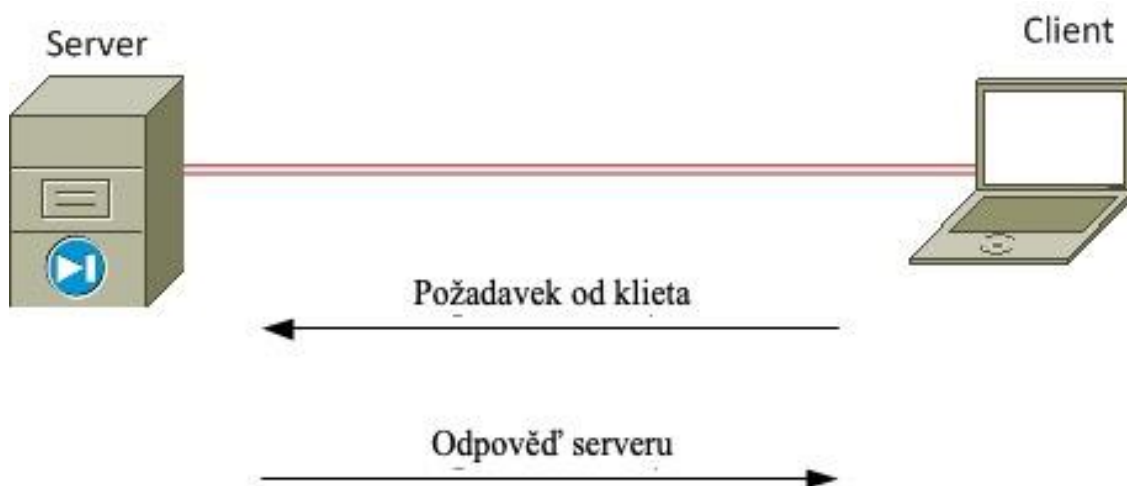
```
1  {
2    "pole": [
3      1,
4      2,
5      3
6    ],
7    "pohlavi": "muz",
8    "vek": 24,
9    "jmeno": "jaroslav",
10   "prijmeni": "evjak",
11   "retezec": "Dobry den"
12 }
```

Obrázek 12: Vzor formátu JSON, který je složen z pole a řetězců.

Struktura, kterou tento formát využívá k ukládání dat má výhodu ve své snadné čitelnosti, ovšem za cenu větší velikosti souboru. To vše bohužel na újmu velikosti dat.

3.1.5 REST API

Webové služby jsou prakticky webové servery, které podporují potřeby webových stránek nebo stránek aplikace. Klientské programy používají rozhraní API pro komunikaci s webovými službami. API vystavuje soubor dat a funkcí, které usnadňují komunikaci mezi počítačovými programy a umožňují jim výměnu informací. Rozhraní API reaguje na požadavek od uživatele a zpět mu vydává odpověď (Obr 13.), (Masse 2012).



Obrázek 13: Znáornění API komunikace

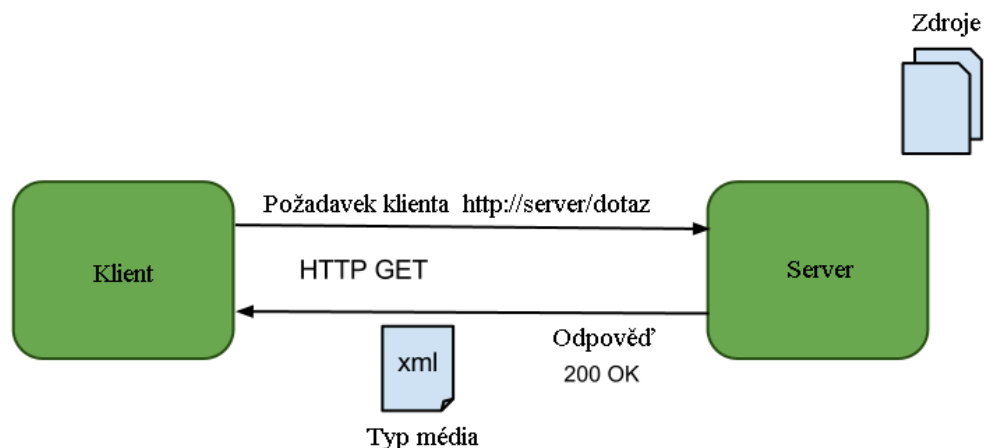
Pojem REST byl poprvé představen v roce 2000, jedná se o architektonický styl, který se běžně používá pro moderní webové služby využívající API rozhraní. Je datově orientován, na rozdíl od známého SOAP, který je orientován procedurálně. Rozhraní REST API znamená, že web používající API odpovídá architektonickému stylu REST.

Webové služby nám definují protokol pro volání vzdálených procedur. REST nám definuje, jak přistupovat k datům. Data zde mohou představovat kromě samostatných dat také stavy dané aplikace (pokud je lze konkrétními daty popsat). Všechny zdroje mají svůj vlastní identifikátor zvaný URI (Uniform Resource Identifier), díky kterému lze ke zdrojům přistupovat pomocí čtyř základních metod, které definuje REST.

Dnešní frameworky pro server-side vývoj velice napomáhají k vytvoření REST rozhraní. Jsou schopné nadefinovat příslušné funkce pro všechny důležité metody, na základě toho je vytvoření RESTful rozhraní jednoduché. Díky vlastnostem tohoto rozhraní se s ním budeme setkávat v budoucnu stále častěji (Masse 2012).

3.1.5.1 GET

Pro přístup ke zdrojům se využívá základní metody **GET**. Pravidelně se s ní setkává denně každý z nás, jedná se o požadavek na danou stránku serveru. Jak bylo řečeno díky tomu, že každý zdroj má vlastní identifikátor, pomocí požadavku **HTTP GET** získáme námi požadovaná data ze zdroje (Obr 14.).



Obrázek 14: Zobrazení REST API komunikace mezi klientem a serverem

3.1.5.2 POST

K vytvoření dat slouží metoda **POST**, je stejně jednoduchá a přímočará jako metoda pro získání dat **GET**. Ve chvíli, kdy se volá metoda POST není znám identifikátor URI, protože v danou chvíli ještě zdroj neexistuje. Po odeslání by server měl vrátit stavový kód, který nám řekne, zda vytvoření dat proběhlo úspěšně nebo ne.

3.1.5.3 DELETE

Kromě výše uvedených metod se dá samozřejmě zdroj i smazat, zavoláním URI http metody DELETE. Volání je podobné jako u funkce GET. V praxi však bývá problémem tuto metodu zavolat, protože HTML formuláře či HTTP nástroje umí pracovat pouze s metodami GET a POST. Tento fakt samozřejmě jde provést jiným způsobem, pomocí metody POST, která v parametru bude sdělovat, že má být použita metoda DELETE.

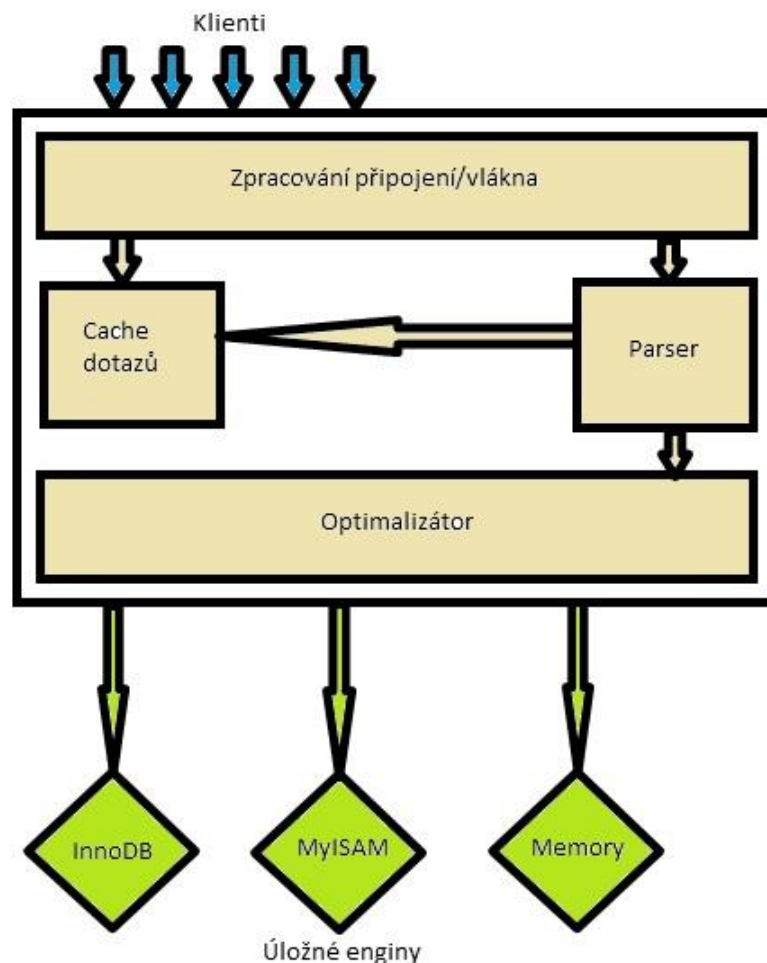
3.1.5.4 PUT (Update)

Metoda je velice podobná metodě POST, hlavním rozdílem je, že zde URI konkrétního zdroje, který chceme změnit už známe. Novou hodnotu předáme v parametru stejně jako u metody POST, také zde platí, že některý nástroj tuhle metodu nepodporuje, a proto se používají jiné náhradní metody, podobně jak u metody DELETE jak tomu bylo u metody DELETE.

3.1.6 MySQL

Jedná se o databázový systém, který je multiplatformní s bezplatnou nebo placenou komerční licenci. Při vývoji byl kladen důraz především na rychlost a díky tomu se některé funkce zjednodušily a jiné odstranily. V dnešní době se jedná o jednu z nejpoužívanějších databází právě díky jejímu výkonu a jednoduché implementaci. Silným stránkou relační databáze je využití cizích klíčů. Cizí klíče se používají pro propojení mezi dvěma tabulkami dle zvoleného sloupce. Komunikace s databází probíhá skrz jazyk SQL.

Architektura MySQL je odlišná od ostatních architektur databázových serverů a je vhodná pro řešení různých úloh (Obr 15), (Nixon, 2018).



Obrázek 15: Architektura MySQL serveru, kde vrchní vrstva obsahuje nástroje pro komunikaci klient/server, druhá vrstva obsahuje funkce, optimalizaci a parsing. Je to v podstatě mozek MySQL, poslední vrstva ukládá a získává data uložená v MySQL.

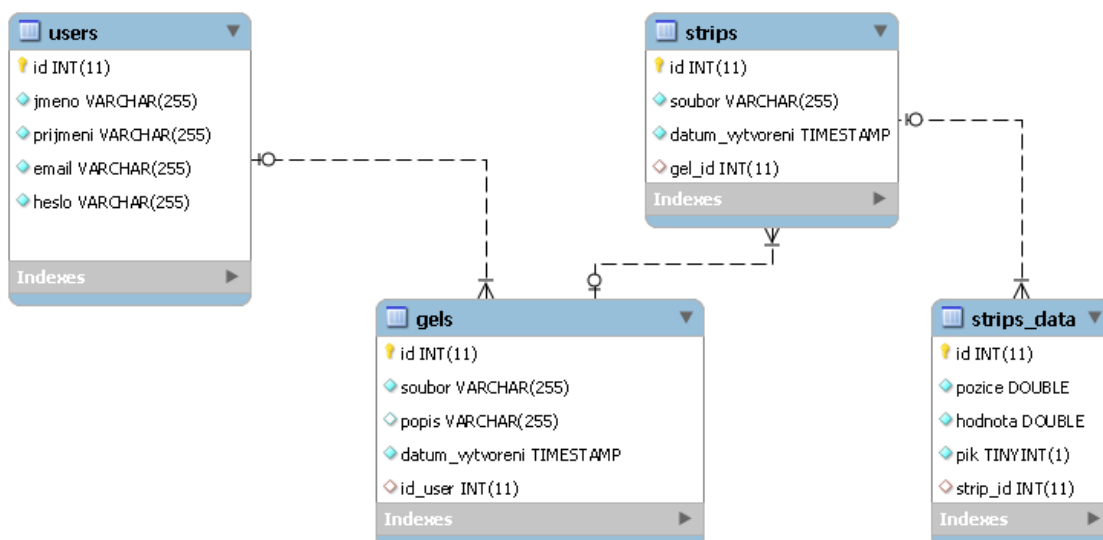
3.2 Návrh databáze SoftGel

Možnou variantou pro SoftGel byla relační databáze MariaDB. Databáze je vyvíjena komunitou a považuje se nástupnickou větví MySQL. Hlavním důvodem vytvoření MariaDB bylo zachování licence svobodného softwaru GNU GPL. Myšlenka pro vytvoření této větve vznikla od původních vývojářů MySQL, kteří se obávali osudu a nasměrování vývoje MySQL po jeho odkoupení společností Oracle.

Nicméně pro SoftGel byla zvolena databáze MySQL, která se v dnešní době řadí mezi nejpoužívanější z SQL databází. Volba byla založena na skutečnosti, že data mají tabulkovou povahu, ale i skrze nutnost zachovat vztahy mezi jednotlivými tabulkami. Tabulky nám zde reprezentují typy záznamů, mezi které patří uživatelé, gely, jednotlivé stripy gelu a hodnoty daného stripu. Vztahy jsou myšleny relace mezi tabulkami, které nám slouží k propojení mezi tabulkami. Jako příklad nám poslouží tabulka gelů, která je navázaná na tabulku jednotlivých stripu a ty na hodnoty stripu. Díky tomu ke každému novému záznamu v tabulce gelů jsou přiřazeny stripy a následně ke stripům odpovídající vypočítané hodnoty

V aktuální alfa verzi databáze SoftGel se nachází čtyři tabulky pro ukládání (Obr. 16):

- **Uživatelé** – Tabulka obsahuje údaje jednotlivých uživatelích, jako je jejich jméno, příjmení, email a heslo pro přihlášení. Tabulka slouží jednak pro přihlášení a dále díky ní lze zjistit jaké gely nahrál uživatel, což je užitečné pro filtrování gelů při výběru pro porovnávání.
- **Gely** – Obsahuje informace o datu vytvoření, popis gelu, názvu souboru a identifikace jaký uživatel gel nahrál
- **Stripy** – Podobně jako gely obsahují datum vytvoření, název stripu a identifikace dle které se přiřadí ke správnému gelu.
- **Data stripů** – Tabulka obsahuje vypočítaná data z jednotlivých stripů, které slouží pro tvorbu grafu a následné porovnání.



Obrázek 16: Schéma databáze a jednotlivých tabulek pro ukládání dat alfa verze softwaru SoftGel.

3.3 Tvorba webového rozhraní

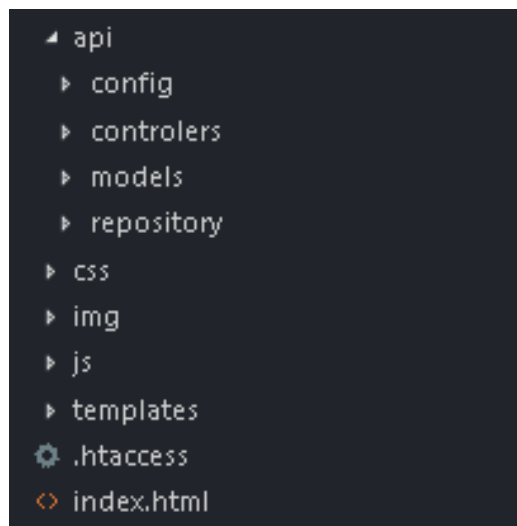
Při implementaci webového rozhraní byla použita kombinace jazyků PHP a JavaScript. Pro vizuální stránku byl použit značkovací jazyk HTML, který se poté ještě graficky upravil pomocí CSS. Aby se předešlo zpomalení serveru, byl zde použit JavaScript, který se od PHP odlišuje tím, že pracuje na straně klienta.

3.3.1 Souborová struktura softwaru

Soubory software jsou standardně organizované do složek podle jejich typu a funkcí. Základem softwaru je rozdělení na frontend část, která se zobrazuje uživateli a backend část jejíž úkolem je logika aplikace a zápis do databáze. Díky nastavení fetchApi v JavaScriptu, který se stará o vykreslování, je URL adresa elegantní. Samostatné vykreslování jednotlivých stránek je založené na tom, že JavaScript si získá informace o volané stránce a překreslí tu původní. Následkem toho je, že se k jednotlivým stránkám nedá přistupovat skrz URL (např. www.softgel.cz/porovnaní).

Logika softwaru je rozdělena na čtyři části, které jsou založeny na principu MVC architektury. Na obrázku číslo 17 je znázorněno jejich strukturní složení:

- **Config** – Zde se nachází základní nastavení pro připojení k databázi (adresa databáze, jméno, heslo a jiné)
- **Controllers** – Obsahují soubory, které na základě parametrů co obdrží vykonají určitou funkci. Jejich hlavní funkcí je zvolení a zavolání správné funkce z repositáře. Pokud se uživatel například zaregistruje, zavolá se userController, který na základě parametru (?action=createUser) zavolá z repositáře funkci pro vytvoření nového uživatele.
- **Repository** – Obsahuje soubory zajišťující komunikaci mezi modelem a kotrolerem. Jejich úkolem je připravit data, které odeslal uživatel pro následné zapsání do databáze. Kromě toho obsahují i výpočetní funkce, například detekci píků.
- **Models** – Zde se nachází soubory pro komunikaci s databází. Získávají data z databáze nebo naopak nahrávají data, které získaly z repositáře.



Obrázek 17: Struktura, která obstarává logickou část softwaru SoftGel.

4 Výsledky a diskuze

4.1 Nahrání gelu

Nahrávání obrázku gelu je jednou z důležitých součástí aplikace. Výsledné obrazy gelů mají k dokonalosti často daleko, a proto je nutné obraz předzpracovat. V aktuální alfa verzi softwaru je pro následnou analýzu potřeba zvolit místa standardů na gelu, k dispozici jsou tři z nejběžnějších standardů.

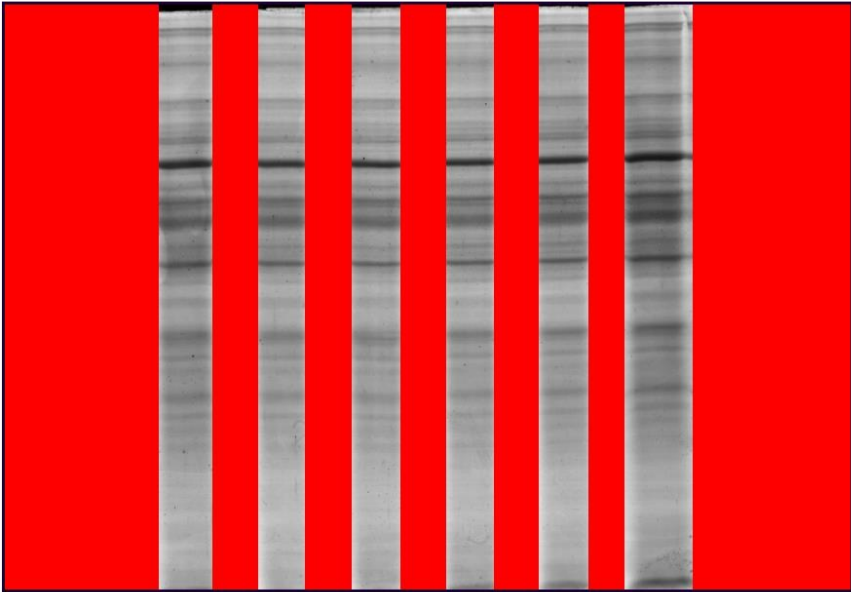
Po nahrání gelu je první částí eliminace šumu, a to kvůli snazší automatické detekci stripů. Díky skutečnosti, že je obraz šedotónový, kde šum (pozadí) má daleko vyšší hodnoty šedi než hledané stripy, je potřeba spočítat průměrnou hodnotu v celém obraze a na základě toho následně šum eliminovat (Zdrojový kód 1).

```
/*
 *
 * Výpočet průměrné šedi ve slupci a obraze
 * @return void
 *
 */
public function avgGray(){
    $celyObrazek =0;
    $pxls = imagesy($this->image) * imagesx($this->image); //Výpočet velikosti obrazu
    for ($x=0; $x < imagesx($this->image); $x++){
        $column = 0;
        for ($y=0; $y < imagesy($this->image); $y++){ // Procházení obrázku po sloupcích
            $color = imagecolorat($this->image, $x, $y); // získání RGB barvy
            $column += ($color >> 16) & 0xFF; //získání jednoho kanálu a následné sčítání
        }
        $celyObrazek += ($column >> 16) & 0xFF;
    }
    array_push($this->avgColumn,round(($column/
    imagesy($this->image)),0,PHP_ROUND_HALF_UP)); // uložení hodnoty sloupce do pole
}
$this->avg = round(($celyObrazek / $pxls),0,PHP_ROUND_HALF_UP); //výpočet
konečné hodnoty šedi pro celý obrázek
}
```

Zdrojový kód 1: Výpočet průměrné hodnoty šedi v obraze.

Dalším krokem po výpočtu průměrné hodnoty je odfiltrování nežádoucího šumu z obrazu. Díky vypočtené průměrné hodnotě se následná eliminace provádí na základě porovnávání hodnoty šedi v pixelu vůči průměru v obraze. Pokud je hodnota pixelu menší (ve smyslu blíže k 0), než průměrná hodnota v obraze, je pixel označen za šum a obarven do červena, jinak zůstává nezměněn. Z ořezaného gelu software ořeže jednotlivé stripy a uloží je do databáze pro následné porovnávání. Výsledný obraz, kde je eliminován šum, je poté zobrazen uživateli (Obr. 18).

Výběr standardu



Standardy	Pozice
21.5	102
31.0	160
45.0	273
66.2	401
97.4	569

Test

Obrázek 18: Výsledek předzpracování a zobrazení ořezaného gelu.

4.2 Detekce píků

4.2.1 Výpočet standardů

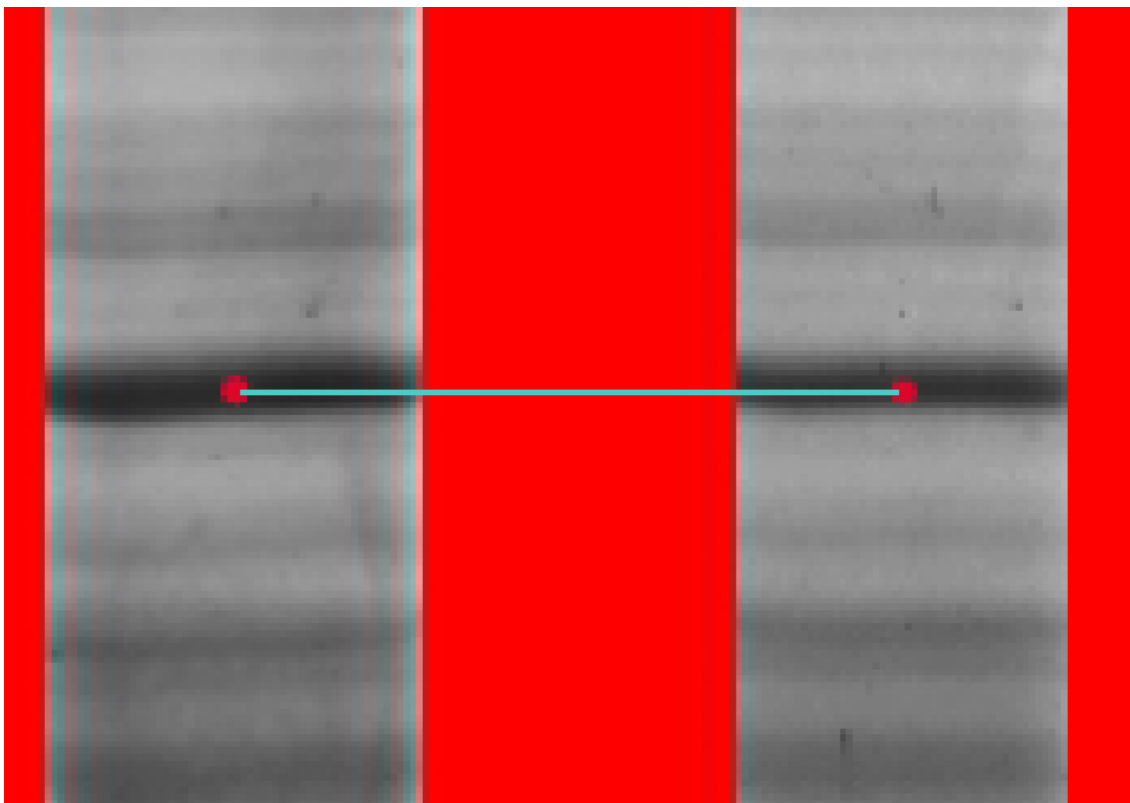
Pro porovnávání jednotlivých stripů je potřeba znát ke každé hodnotě z obrazu (pixelu) příslušnou hodnotu standardu. Díky skutečnosti, že software nezná všechny hodnoty standardů, ale pouze uživatelem určené, je potřeba tyto hodnoty dopočítat pomocí ekvidistantní sítě uzlů a interpolace.

Ekvidistantní síť uzlů nám zajistí rovnoměrné rozložení hodnot v grafu, díky kterým je porovnávání přesnější a vizualizace příjemnější. Pro interpolaci software využívá **Newtonův algoritmus**. Vypočítané hodnoty jsou zaokrouhlovány na tři desetinná místa., díky tomu je následné porovnání přesnější, (Zdrojový kód 2).

```
*
*Vytvoření polynomu pro dopočítání standardů
*@return void
public function buildPolynomial($data)
{
    $t = array(); // první pomocné pole
    $a = array(); // druhé pomocné pole
    $keys = $data->pos; //načtení pozit
    $values = $data->values;
    for( $i=0; $i<count( $keys ); $i++ ) // zjištění dostupných hodnot
    {
        $t[$i] = $values[$i];
        for( $j=$i-1; $j>=0; $j-- )
            $t[$j] = ( $t[$j+1] - $t[$j] ) / ( $keys[$i] - $keys[$j] ); //tvorba polynomu
        $a[$i] = $t[0];
    }
    $this->polynomial = $a;
}
*
*Newtonova interpolace pro výpočet standardů
*@return double
*
*/
public function interpolate( $data, $y)
{
    $keys = $data->pos ;
    $n = count( $keys );
    $p = $this->polynomial[$n-1];
    for( $i=$n-2; $i>=0; $i-- )
        $p = $p * ( $y - $keys[$i] ) + $this->polynomial[$i]; //dopočítání standartu pro daný
pixel
    return round($p,3); // zaokrouhlení výsledku na tři desetinná místa
}
}
```

Zdrojový kód 2: Interpolace pomocí Newtonova algoritmu pro výpočet standardů

Z oblastí, které jsou označeny za potencionální následně vyfiltruje maxima. Filtrování je založeno na tom, že se provede ve zkoumaném bodě kontrole levého a pravého okolí. Software nemá specifikovanou toleranci, tudíž porovnává zkoumaný pixel s pixelem vlevo a vpravo. Pokud má bod nižší hodnoty šedi než okolní pixely, je prohlášen za pík (Obr. 20), (Zdrojový kód 3).



Obrázek 20: Zobrazení souhlasných píků se stejným standardem mezi stripy.

```

*
*Detekování píků True/False
* @return array
*/
public function detectPeaks($data){
    $interpol = array(array()); // pole pro data z interpolace
    $stripId = 0;
    $j = 0;
    $peaks = array(array()); // pole pro detekované píky
    $konst = 0.67; // konstanta pro detekci píku
    $this->buildPolynomial($data); // vytvoření polynomu
    foreach($this->stripsPixel as $x){ // procházení pole dle středů stripů
        $peaksLimit = round($this->avgColumnPixel[$j] * $konst); //
        $peaks[$stripId] = [];
        for ($y=0; $y < imagesy($this->image); $y++) { // procházení sloupce obrázku
            $interpol[$stripId][]=$this->interpolate($data, $y); // výpočet interpolace pro
standardary
                $color = imagecolorat($this->image, $x, $y); // zjištění barvy pixelu
                $r = ($color >> 16) & 0xFF; // zjištění RGB pixelu
                $this->pixelValues[$stripId][]=$r; //plnění pole hodnotami barvy pixelu
                if($r < $peaksLimit){ // podmínka zda je pixel v oblasti peaku či ně
souseda
                    if($y != 0 && $y+1 != imagesy($this->image)){
souseda
                        $colorUp = imagecolorat($this->image, $x, $y+1); // zjištění barvy pravého
souseda
                            $colorDown = imagecolorat($this->image, $x, $y-1); // zjištění barvy levého
                                $pxUp = ($colorUp >> 16) & 0xFF; // převod do RGB
                                $pxDown = ($colorDown >> 16) & 0xFF; // převod do RGB
                                    if(($r<=$pxUp && $r< $pxDown) || ($r<$pxUp && $r <= $pxDown)){ //
podmínka pro ověření maxima.
                                        if(count($peaks[$stripId])>0)
                                        {
                                            if(end($peaks[$stripId]) != $r) //pokud je platná podmínka uloží se true
do pole
                                                $peaks[$stripId][]=true;
                                                else{
                                                    $peaks[$stripId][]=false;
                                                }
                                            }else{
                                                $peaks[$stripId][]=true;
                                            }
                                        }
                                        else{
                                            $peaks[$stripId][]=false;
                                        }
                                    }
                                }
                            }
                        }
                    }
                }
            }
        }
    }
}

```

```

$stripId += 1;
$j += 1;
$peaksLimit = 0;
}
$complete = []; // pole pro výsledek
for ($x=0; $x<sizeof($interpol); $x++){ // procházení pole interpolace
    $completeLine = [];
    for ($i=0; $i < sizeof($interpol[$x]); $i++) { //procházení pole interpolace
        $line = [$interpol[$x][$i], $this->pixelValues[$x][$i], $peaks[$x][$i]];
        zapisování výsledných hodnot do pole
        array_push($completeLine,$line);
    }
    array_push($complete,$completeLine);
}
return $complete;
}

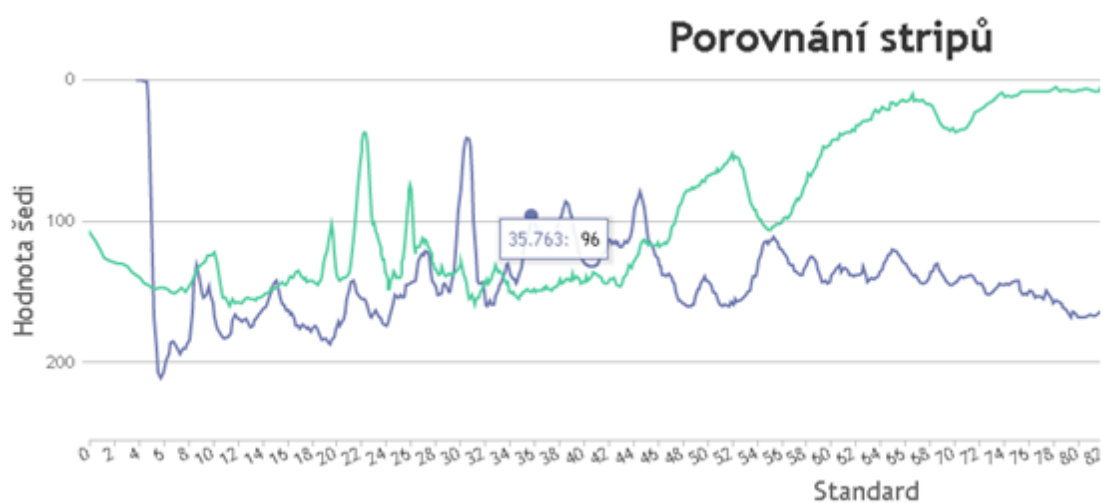
```

Zdrojový kód 3: Dupočítání standardů s následnou detekci píků

4.3 Grafické porovnání stripů

Aplikace SoftGel je vytvořena pro porovnávání výsledků z gelové elektroforézy a funguje na všech moderních webových prohlížečích. SoftGel zvládá porovnávat libovolné množství stripů mezi sebou ve velice krátkém čase.

Samostatné porovnávání nabízí mnoho nástrojů. Graf je možné například přiblížit pro lepší orientaci, zobrazuje po najetí myší přesně dané hodnoty určitému standardu. Mezi další nástroje patří přepnutí mezi jednoduchým nebo detailnějším grafem a grafem, který zobrazí pouze píky. Je možný také export výsledků do formátu JPEG nebo PNG (Obr. 21).



Obrázek 21: Grafická znázornění dvou stripů odlišných gelů. Graf po najetí myší zobrazuje hodnotu standardu a šedi.

5 Závěr

Cílem této práce bylo zpracování literární rešerše o dostupných řešeních pro analýzu skenu gelu a následně zasvěcení do technického pozadí. Dále vytvoření webové aplikace pro porovnávání skenu gelů, kde jejím hlavním úkolem byla možnost porovnání libovolného počtu stripů s grafickým výstupem. Tímhle se od mnohých řešení liší, protože většina softwarů umožňuje analýzu pouze jednoho skenu gelu. Aplikace mimo to měla umožnit u nahraného gelu automaticky eliminovat šum, detekovat jednotlivé stripy, oblasti možných výskytů píky a samostatné píky.

V druhé kapitole popisují principy gelové elektroforéze. Současně zde nastiňují problémy s výsledky z gelové elektroforézy, které bývají často porušeny kvůli nedodržování postupů či jiným vlivům. To má za následek následné zhoršení výsledků počítačové analýzy. Dále popisují existující softwary, které dokážou zkusitelně eliminovat, ke kterým patří GelViewer a UN-SCAN-IT. Následně seznamují s tím, jak počítač vnímá dané skeny gelů.

Třetí kapitola je již praktická a popisuje použité technologie při tvorbě softwaru SoftGel. Vysvětlují, jakým způsobem funguje REST API, na kterém je software založený. Dále poukazují strukturu samostatného softwaru.

Poslední kapitola je samostatný vývoj softwaru, kde popisují, jak funguje ořezání gelu, pomocí jakého algoritmu jsme schopni dopočítat chybějící standardy a software vyhodnocuje píky a zobrazuje výsledná data.

Do budoucna je v plánu rozšířit funkce pro ořezání skenu a následné porovnávání. Ke kterým patří například možnost eliminovat zkusitelně obrazu, vylepšení algoritmu pro detekci píků a možnost určení konstanty šumu uživatelem.

6 Literatura

- Aberham C., Weber S., Phares W. (1996): Spontaneous in the human immunodeficiency virus type 1 gag gene that affect viral replication in the presence of cyclosporins. *Journal of Virology* 70, 3536-3544.
- Ater T. (2017): *Building progressive web apps: bringing the power of native to the browser*. Sebastopol, CA: O'Reilly Media, 268 stran.
- Bassett L., (2015): *Introduction to Javascript Object Notation: a to-the-point guide to JSON*. Sebastopol, CA: O'Reilly.
- Berčík J., [] (1977): *Fyzikálne a fyzikálno-chemické analytické metódy*. 1. vyd. Bratislava: ALFA, 503 stran.
- Digimanie.cz: <https://www.digimanie.cz/reprezentace-barev-v-pc-rgb-a-barevny-prostor/1653> (11. 5. 2019)
- Gaš B., (2001): Kapilární elektroforéza: Separační analytická metoda pro věk mikročipů. *Vesmír* 7, 1214-4029.
- GelAnalyzer : <http://www.gelalyzer.com/> (11. 5. 2019).
- Gutmans A., Bakken S. Rethans D. (2015): *PHP5 Power Programming*. PrenticeHall, New Jersey, U.S.A.
- Chrambach, A., Rodbard D., (1971): Polyacrylamide Gel Electrophoresis, Vol 172, 3982, 440-451.
- Jedra, M., El Khattabi, N., Limouri, M. and Essaid, A. (2019): Recognition of seed varieties using a time-delay neural network: analysis of electrophoretic images, Vol 22,1, 1-10
- Jeppson J. O., Laurell C. B., Franzén B., (1979). Agarose gel electrophoresis. *Clinical Chemistry*, 629-638.
- Kealey, D., (2001): *Bios Instant Notes In Analytical Chemistry*. Taylor & Francis. Oxford: BIOS, 342 stran.
- Klouda P. (1996): *Moderní analytické metody: Učebnice základů instrumentálních analytických metod*. 1. vyd. Ostrava: Nakladatelství Pavel Klouda, 203 stran.
- Kouranov A., Xie L., Cruz J., Chen L., Westbrook J., [] (2006): The RCSB PDB information portals for structural genomics. *Nucleic Acids Research*, Vol.34, Isuppl_1, 302–305. <https://doi.org/10.1093/nar/gkj120>
- Kozma D., Simon I., Tusnády E. G. (2012): PDBTM: Protein Data Bank of transmembrane proteins after 8 years. *Nucleic Acids Research*, Vol. 41, 1, 524–529. <https://doi.org/10.1093/nar/gks1169>
- Massé, M., (2011): *Rest Api Design Rulebook*. O'reilly Media.
- Nixon, N., (2018): *Learning Php, Mysql & Javascript: With JQuery, Css & Html5*, O'reilly Media.

- Pilgrim M. (2010): HTML5: Up and Running: Dive into the Future of Web Development, O'ReillyMedia, Inc., Sebastopol, California, U.S.A
- Pouchlý J., (2008): Fyzikální chemie makromolekulárních a koloidních soustav. Vyd. 3. Praha: Vydavatelství VŠCHT, 205 stran.
- Silkscientific: <https://www.silkscientific.com/gel-analysis.htm> (11. 5. 2019)
- Storchová Z., (1998): Jak to udělat, aby molekula byla dobře viditelná. Vesmír 7
- Žára O. (2015): JavaScript. Computer Press, Brno, ČR.
- CanvasJS: <https://canvasjs.com/> (11. 5. 2019)