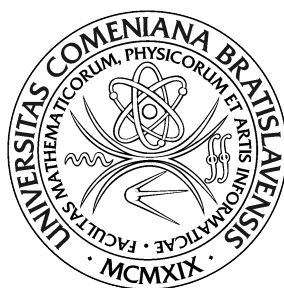


UNIVERZITA KOMENSKÉHO V BRATISLAVE
FAKULTA MATEMATIKY, FYZIKY A INFORMATIKY



KLABORATÍVNY GRAFICKÝ EDITOR PRE MEDIAWIKI

Diplomová práca

2018

Bc. Martin Krasňan

UNIVERZITA KOMENSKÉHO V BRATISLAVE
FAKULTA MATEMATIKY, FYZIKY A INFORMATIKY



KLABORATÍVNY GRAFICKÝ EDITOR PRE MEDIAWIKI

Diplomová práca

Študijný program: Aplikovaná informatika
Študijný odbor: 2511 Aplikovaná informatika
Školiace pracovisko: Katedra aplikovanej informatiky
Školiteľ: doc. RNDr. Zuzana Kubincová, PhD.

Bratislava, 2018

Bc. Martin Krasňan



Univerzita Komenského v Bratislave
Fakulta matematiky, fyziky a informatiky

ZADANIE ZÁVEREČNEJ PRÁCE

Meno a priezvisko študenta: Bc. Martin Krasňan
Študijný program: aplikovaná informatika (Jednoodborové štúdium, magisterský II. st., denná forma)
Študijný odbor: aplikovaná informatika
Typ záverečnej práce: diplomová
Jazyk záverečnej práce: slovenský
Sekundárny jazyk: anglický

Názov: Klaboratívny grafický editor pre MediaWiki
Collaborative graphics editor for MediaWiki

Cieľ: Navrhnuť a implementovať grafický editor pre MediaWiki určený pre žiakov umožňujúci kolaboratívne kreslenie a úpravu obrázkov. Vytvorený editor integrovať s wiki.matfyz.sk.

Vedúci: doc. RNDr. Zuzana Kubincová, PhD.
Konzultant: Mgr. Ján Kľuka, PhD.
Katedra: FMFI.KZVI - Katedra základov a vyučovania informatiky
Vedúci katedry: doc. RNDr. Zuzana Kubincová, PhD.
Dátum zadania: 06.10.2016

Dátum schválenia: 13.10.2016
prof. RNDr. Roman Ďurikovič, PhD.
garant študijného programu

študent

vedúci práce

Čestne vyhlasujem, že som túto diplomovú prácu vypracoval samostatne pod vedením doc. RNDr. Zuzany Kubincovej, PhD., s použitím zdrojov uvedených v zozname použitej literatúry.

Bratislava, 2018

.....
Bc. Martin Krasňan

Pod'akovanie

...podakovanie...

Abstrakt

KRASŇAN, Martin: *Klaboratívny grafický editor pre MediaWiki* [Diplomová práca]. - Univerzita Komenského v Bratislave. Fakulta matematiky, fyziky a informatiky; Katedra aplikovanej informatiky. - Školiteľ: doc. RNDr. Zuzana Kubincová, PhD.. Bratislava: Bratislava, 2018. ?? strán.

Text abstraktu...

Kľúčové slová: klucove, slova, sk, ...

Abstract

KRASŇAN, Martin: *Collaborative graphics editor for MediaWiki* [Master thesis]. - Comenius University in Bratislava. Faculty of Mathematics, Physics and Informatics; Department of Applied Informatics. - Supervisor: doc. RNDr. Zuzana Kubincová, PhD.. Bratislava, 2018. ?? pages.

Text of abstract

Keywords: keywords, en, ...

Obsah

1	Úvod	1
2	Motivácia	2
3	Prehľad problematiky	3
3.1	MediaWiki	3
3.2	Prehľad technológií	5
3.2.1	PHP	5
3.2.2	HTML	5
3.2.3	Kaskádové štýly (CSS)	6
3.2.4	JavaScript	6
3.2.5	JavaScriptová knižnica Node.js	7
3.2.6	JavaScriptová knižnica Less.js	7
3.2.7	JavaScriptová knižnica Fabric.js	7
3.2.8	JavaScriptový framework Angular.js	8
3.2.9	JavaScriptová knižnica Socket.io	8
3.2.10	WebSocket	9
3.3	Rozšírenia MediaWiki	10
3.3.1	ResourceLoader	11

<i>OBSAH</i>	ix
3.4 Cieľ práce	13
4 Predchádzajúce riešenia	14
4.1 SVGEdit	14
4.2 Figma	15
5 Návrh modelu	17
6 Implementácia	18
7 Výsledky	19
8 Záver	20

1

Úvod

...

2

Motivácia

...

3

Prehľad problematiky

3.1 MediaWiki

MediaWiki je publikačný systém určený na jednoduchú a efektívnu tvorbu obsahu webových stránok založený v roku 1995 Howardom G. Cunninghamom. Tento systém je voľne šíriteľný Open Source softvér, licencovaný pod GNU (General Public Licence), pričom jeho zdrojové kódy sú napísane v jazyku PHP. Na uchovávanie dát využíva relačnú databázu MySQL. Obsah stránky je vytváraný dynamicky, pomocou wikitextu.

Wikitext je vlastný značkovací jazyk, ktorý zabezpečuje jednoduchú edi-

táciu obsahu webových stránok bez nutnosti ovládať HTML značkovací jazyk a CSS kaskádové štýly. Wikitext je možné písať priamo pomocou spomínanej wikitext syntaxe, kde je možné nadpis sekcie zapísať ako `=Nadpis sekcie=`, podnadpis sekcie `==Podnadpis sekcie==`. Odkazy na iné stránky sa zapisujú pomocou hranatých zátvoriek `[[Názov článku]]`. Podobný zápis sa využíva aj na vkladanie súborových príloh ako napríklad obrázkov, kde je syntax zápisu nasledovná: `[[File:NazovSuboru.jpg]]`. Ďalšie informácie ohľadom spôsobu zápisu wikitextu je možné nájsť v dokumentácii Media-Wiki systému [Med17a]. Obsah stránky je možné zapisovať priamo pomocou tohoto značkovacieho jazyka, alebo pomocou editora formátovaného textu (rich-text editora).

Zmeny vykonané používateľom na obsahu stránky, sú pri ukladaní zapisované do databázy ako samostatná verzia. Históriu zmien (verzií) je možné zobrazíť a v prípade potreby taktiež obnoviť. Pri každej verzii je uchovávaný autor danej zmeny, s dátumom kedy zmenu vykonal.

3.2 Prehľad technológií

3.2.1 PHP

PHP (PHP: Hypertext Preprocessor) je populárny open source skriptovací jazyk, ktorý sa používa najmä na programovanie klient-server aplikácií (na strane servera) a pre vývoj dynamických webových stránok.

PHP bolo inšpirované jazykmi podporujúcimi procedurálne programovanie. Najviac vlastností prebralo od jazyka C a jazyka Perl. V neskorších verziách bolo rozšírené o možnosť používať objekty.

Jedna zo zaujímavých vlastností PHP je, že umožňuje oveľa viac ako bežný skriptovací jazyk. Vďaka modulárnemu návrhu možno PHP používať aj na vývoj aplikácií s užívateľským rozhraním (GUI).

PHP dokáže spolupracovať s relačnými databázami, ako napríklad MySQL, Oracle, IBM DB2, Microsoft SQL Server, PostgreSQL alebo SQLite, pričom si stále zachováva jednoduchú a priamočiaru syntax. PHP beží na takmer všetkých najrozšírenejších operačných systémoch, vrátane UNIXu, Linuxu, Windows či Mac OS X. Spolupracuje s najrozšírenejšími webovými servermi.

3.2.2 HTML

Hypertextový značkový jazyk (HyperText Markup Language; HTML) je značkový jazyk určený na vytváranie webových stránok a iných informácií zobraziteľných vo webovom prehliadači. HTML kladie dôraz skôr na prezentáciu informácií (odseky, fonty, váha písma, tabuľky atď.) ako na sémantiku (význam slov). Pôvodne bol určený ako veľmi zjednodušená podmnožina jazyka SGML, ktorý sa používa v organizáciách s komplexnými publikačnými požiadavkami, ale neskôr sa stal samostatným štandardom (ISO/IEC 15445:2000).

Špecifikáciu jazyka HTML udržiava World Wide Web Consortium (W3C). Do príchodu HTML5 W3C plánovalo nahradiť HTML jazykom XHTML, implementáciou jazyka XML, ktorá mala zjednodušiť prácu prehliadačov aj tvorcov web stránok. V mediawiki sa HTML používa na vykreslenie vizuálnych prvkov.

3.2.3 Kaskádové štýly (CSS)

Kaskádové štýly alebo CSS (skratka z angl. Cascading Style Sheets) je všeobecné rozšírenie (X)HTML. Konzorcium W3C[1] označuje CSS ako jednoduchý mechanizmus na vizuálne formátovanie internetových dokumentov.

Štýly umožnili oddeliť štruktúru HTML alebo XHTML od vzhľadu. Prvá verzia CSS (CSS level 1) vznikla už v roku 1996 a umožňovala prácu s písmami, okrajmi a farbami. V roku 1998 bola doplnená o nové možnosti a vznikol CSS level 2. V súčasnosti je podporovaná vo všetkých novších verziách prehliadačov (Internet Explorer, Opera, Mozilla, Netscape, Safari). Aktuálna verzia je CSS level 3.

3.2.4 JavaScript

JavaScript, je skriptovací programovací jazyk. Jazyk je používaný najmä pri tvorbe webových stránok. Pôvodne ho vyvíjal Brendan Eich zo spoločnosti Netscape Communications pod názvom Mocha, neskôr pod menom LiveScript. Pred uvedením na verejnosť bol premenovaný na „JavaScript“, najmä pre vtedajšiu popularitu jazyka Java. Aj na základe jeho názvu je rozšírený názor, že syntax Javascriptu sa podobá Jave, v skutočnosti bol jeho tvorca najviac inšpirovaný jazykom Self.

3.2.5 JavaScriptová knižnica Node.js

Node.js je open-source medziplatformové run-time prostredie pre vykonávanie JavaScript kódu na strane serveru. Historicky bol Javascript určený hlavne pre programovanie na klientskej strane, kde sú skripty písané v Javascrpite priamo volané v HTML súbore webovej stránky. Kód bol spúšťaný enginom webového prehliadača. Node.js umožňuje použitie JavaScriptu na strane servera a kód sa spúšťa predtým ako je dynamický obsah stránky poslaný do webového prehliadača používateľa. Vďaka tejto výhode sa z Node.js stal jeden zo základných elementov paradigmy "Javacript everywhere"ktorý pri vývoji webových aplikácii umožňuje zjednotenie programovacích jazykov namiesto spoliehania sa na iné programovacie jazyky nutne pre serverovú časť aplikácie.

3.2.6 JavaScriptová knižnica Less.js

Less je CSS pre=procesor, čo znamená, že rozširuje jazyk CSS, pridáva funkcionlitu ktorá umožňuje definovanie premenných, funkcií a množstvo ďalších techník. Vďaka tomu je možné vytvoriť kaskádové štýly, ktoré sú prehľadné, rozšíriteľné s možnosťou jednoduchej zmeny veľkého množstva faktorov ovplyvňujúcich vzhľad aplikácie. Zdrojové súbory v jazyku Less sú kompilované pomocou viacerých možných spôsobov. Napríklad priamo v prehliadači používateľa, v prostredí Node, PHP, .Net a ďalších. Výsledkom kompilácie je vytvorenie súboru s kaskádovými štýlmi vo formáte css.

3.2.7 JavaScriptová knižnica Fabric.js

Fabric.js je výkonná a jednoduchá JavaScriptová knižnica na prácu s grafickou plochou canvas. Poskytuje interaktívny model objektov na prácu s can-

vas elementami. Obsahuje taktiež parser na prácu s SVG grafickými súbormi, vďaka čomu je možné renderovať obsah SVG súboru do grafickej plochy canvas. Canvas síce umožňuje vytváranie objektov jednoduchých tvarov pomocou svojich natívnych funkcií, avšak následná manipulácia s týmito objektami je veľmi ťažkopádna. Fabric.js preto vytvoril sadu príkazov na jednoduchú manipuláciu s grafickými objektami. Stará sa o renderovanie a udržiavanie aktuálneho stavu plochy.

3.2.8 JavaScriptový framework Angular.js

AngularJS je JavaScriptová voľne šíriteľná knižnica (framework), ktorá je zastrešovaná, vyvíjaná a udržiavaná prevažne zamestnancami spoločnosti Google.

Princíp fungovania knižnice spočíva v tom, že po načítaní HTML stránky, do ktorej sú vložené vlastné značky atribútov, ktoré angular interpretuje ako smerníky naviazané na model. Ten je reprezentovaný štandardnými premennými jazyka JavaScript. Hodnoty týchto premenných je možné nastaviť manuálne v rámci kódu aplikácie, alebo získať zo statických alebo dynamických JSON súborov. Framework prispôsobuje a rozširuje tradičné HTML tak, aby prezentoval dynamický obsah prostredníctvom obojsmernej väzby dát, ktorá umožňuje automatickú synchronizáciu modelov a šablón.

3.2.9 JavaScriptová knižnica Socket.io

Socket.IO je JavaScriptová knižnica vytvorená pre realtime webové aplikácie. Socket.IO umožňuje realtime obojstrannú komunikáciu medzi webovým klientom a servermi. Socket.IO má dve časti: - klientská strana ktorá beží vo webovom prehliadači a serverovú časť pre node.js. Obe časti majú takmer identické API. Podobne ako node.js je socket.io "event-driven", teda funciona-

lita je vyvolávaná pomocou volaní definovaných funkcií ktoré sa volajú pri určitých akciach používateľa alebo systému.

Socket.io primárne používa WebSocket protokol. Aj keď by to mohol byť použitý len ako wrapper pre WebSocket, socket.io ponúka veľa dodatočných funkcií vrátane vysielania na viacero socketov, ukladanie dát pre každého klienta a asynchrónne vstupno/výstupné operácie. Socket.io v mojej implementácii mediawiki je inštalovaný pomocou balíčkového manažéra npm.

3.2.10 WebSocket

WebSocket je počítačový komunikačný protokol poskytujúci plny duplexný (obojsmerný) komunikačný kanál cez jedno TCP pripojenie. Protokol WebSocket bol štandarizovaný komisiou IETF ako RFC 6455 v roku 2011 a WebSocket API pre webové IDL bolo štandarizované konsorciom W3C.

WebSocket bol navrhnutý tak, aby mohol byť spúšťaný vo webových prehliadačoch a na webových serveroch, kde môže byť použitá ľubovoľná klientská a serverová aplikácia. WebSocket je nezávislý protokol založený na základe TCP. Jeho jediný vzťah ku HTTP je, že jeho handshake (naviazanie spojenia) je interpretované HTTP servermi ako požiadavka na aktualizáciu. Protokol WebSocket umožňuje interakciu medzi prehliadačom a webovým serverom s nižšou réžiou, uľahčuje real-time prenos dát (prenos dát v reálnom čase) zo servera na server. To je možné preto, že poskytuje štandarizovaný spôsob pre odosielanie obsahu zo serveru do prehliadača bez toho aby to bolo na požiadavku klienta (napríklad ako v AJAX volaní) a umožňuje predanie správ tam a naspäť pokiaľ je udržané otvorené spojenie. Týmto spôsobom môže prebiehať obojsmerná komunikácia medzi webovým prehliadačom a serverom. Táto komunikácia prebieha klasicky cez TCP port s číslom 80 alebo s portom 443 v prípade ak sa jedná o šifrované TLS spojenie. Toto

je výhoda v prostrediach kde sú webové pripojenia k internetu blokované firewallom. Podobne obojsmernej komunikácie prehliadač-server bolo v minulosti dosiahnutej pomocou neštandardizovanými spôsobmi ako napríklad knižnica Comet.

Protokol WebSocket je v súčasnosti podporovaný vo väčšine webových prehliadačov. WebSocket ale taktiež vyžaduje podporu na strane serverom a webových aplikácií.

3.3 Rozšírenia MediaWiki

Systém MediaWiki ja navrhnutý spôsobom, ktorý umožňuje pridávať funkcionality pomocou rozšírení (angl. Extensions). Rozšírenia sú časti zdrojového kódu, ktorý môže ovplyvniť vizuálne prvky stránky alebo funkčné vlastnosti systému. Vzhľadom na kategóriu rozšírenia sa dajú využiť napríklad na:

- rozšírenie wikitext syntaxe o vlastné prvky (Parser Extensions Category)
- pridanie schopnosti reportovania informácií (Special page Extensions Category)
- zmena vzhľadu MediaWiki (User interface Extensions)
- zvýšenie zabezpečenia pomocou vlastných autentifikačných mechanizmov (Authentication and Authorization Extensions Category)

Zoznam kategorizovaných rozšírení je možné nájsť na stránkach tvorcov MediaWiki systému. Množstvo rozšírení je vytvorených a udržiavaných samotnými autormi systému, avšak vzhľadom na to, že zdrojové kódy sú voľne dostupné, veľa ich pochádza od vývojárov tretej strany.

Rozšírenia môžu byť do systému inštalované iba s administrátorským prístupom do súborového systému na serveri. Inštalácia pozostáva zo skopírovania zdrojových súborov do adresára `$IP/extensions/nazov_rozsirenia/`. Následne ho je možné inicializovať v koreňovom adresári MediaWiki inštalácie, v súbore `LocalSettings.php`, pomocou globálnej funkcie `wfLoadExtension('nazov_rozsirenia')`.

Pre správne fungovanie rozšírenia je potrebné zabezpečiť niekoľko kľúčových vlastností, pozostávajúcich z týchto krokov:

- Zaregistrovať akýkoľvek media handler, parsovaciu funkciu, špeciálne stránky, vlastné XML značky a premenné použité vašim rozšírením.
- Definovať a zvalidovať každú konfiguračnú premennú ktorá je zadefinovaná vo vašom rozšírení.
- Pripraviť triedy použité vo vašom rozšírení pre autonačítanie.
- Rozhodnúť ktoré časti inštalačného nastavenia majú byť vykonané okamžite a ktorá majú čakať pokiaľ sa inicializuje jadro MediaWiki.
- Definovať dodatočné hooky potrebné pre rozšírenie.
- Vytvoriť / skontroluje všetky nové databázové tabuľky nutné pre rozšírenie.
- Nastaví lokalizáciu vášho rozšírenia.

3.3.1 ResourceLoader

ResourceLoader je tzv. "delivery system" pre optimalizáciu načítavania a maňazovania modulov.

Každá stránka postavená na MediaWiki obsahuje stovky kilobajtov Javascriptu. Vo veľa prípadoch sa niektoré časti tohto kódu nevyužívajú z dôsledku používania nepodporovaného prehliadača alebo sa kód na danej stránke nepoužíva. V týchto prípadoch je čas načítania zbytočne dlhší ako je potrebné.

ResourceLoader tento problém rieši načítavaním zdrojových kódov "resources" na vyžiadanie a len na prehliadačoch ktoré ho podporujú. Tento proces pozostáva z nasledujúcich bodov:

1. Minifikácia a zjednotenie

- Redukuje veľkosť kódu a čas na jeho parsovanie alebo sťahovanie.
- Javascript a CSS súbory sú načítavané v jednej špeciálne formátovanej ResourceLoader Implement "serverovej odpovede".

2. Načítavanie v dávkách

- Redukuje počet serverových volaní.
- Serverová odpoveď pre načítanie podporuje načítanie viacerých modulov na jedno volanie, ktoré už obsahuje minifikované a spojené súbory.

3. Data URI vkladanie

- Ešte viac redukuje počet požiadavok na server, čas odozvy a využitie pásma.
- Obrázky referencované v CSS môžu byť vložené ako data URI linky.

3.4 Cieľ práce

- Navrhnuť prostredie grafického editora
- Implementovať grafický editor do prostredia MediaWiki pomocou rozšírenia
- Navrhnuť a implementovať spôsob ukladania verzií výstupných súborov grafického editora
- Navrhnuť a implementovať komunikačný protokol na synchronizáciu grafického editora pre viacero nezávislých používateľov
- Integrovať rozšírenie s webovou stránkou fakulty <http://wiki.matfyz.sk>

4

Predchádzajúce riešenia

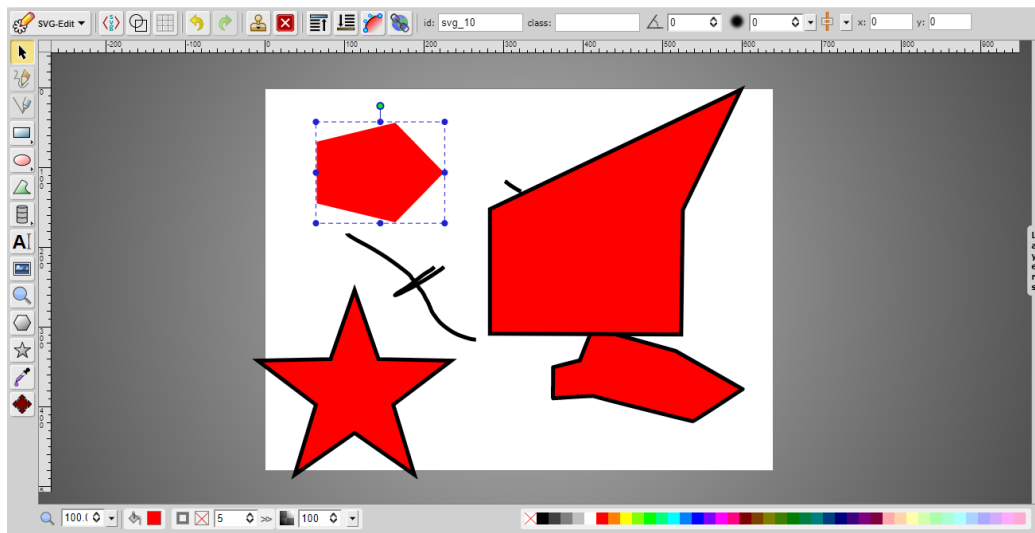
V nasledujúcej kapitole opíšeme existujúce riešenia, zameriame sa na ich výhody a prípadné nevýhody.

4.1 SVGEdit

SVGEdit je jediné oficiálne rozšírenie existujúce v dobe písania tejto diplomovej práce. Aktuálne ja v stave experimental, čo naznačuje že nie všetka funkcionality bude bezproblémová. Poskytuje možnosti vytvárania a editovania súborov vektorovej grafiky vo formáte svg. Využíva open source SVG-edit

widget ktorý poskytuje štandardnú funkcionality vektorového editora, s relatívne vysokou kvalitou spracovania.

Nevýhodou tohoto riešenia je, že samotný editor je vložený formou iframe widgetu zo stránok vývojárov tohoto widgetu. To spôsobuje nemožnosť upraviť vzhľad alebo funkcionality editora. Prepojenie s MediaWiki systémom je zabezpečené pomocou asynchrónnych ajax volaní. Editor taktiež nepodporuje kolaboratívnu úpravu jedného súboru viacerými používateľmi. Ďalšou nevýhodou je jeho zložité ovládanie ktoré sa nehodí pre cieľovú skupinu finálnej aplikácie, ktorou sú študenti základných a stredných škôl.



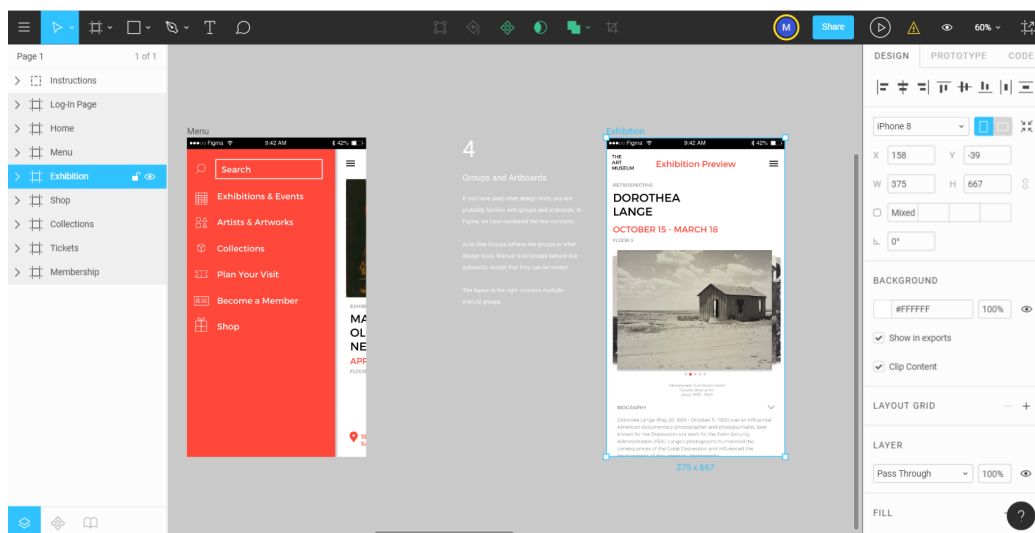
Obr. 4.1: Prostredie editora SVG-edit

4.2 Figma

Figma je profesionálny grafický editor s možnosťou kolaboratívnej práce viacerých používateľov. Je určený predovšetkým pre používateľov zaoberajúcich sa grafickým návrhom mobilných webových alebo desktopových aplikácií. Jeho hlavnou výhodou je intuitívne ovládanie, realtime komunikácia medzi

spolupracujúcimi používateľmi a bohatá funkcionálnosť.

Keďže sa však jedná o samostatnú webovú aplikáciu, nie je možné ju implementovať do MediaWiki systému.



Obr. 4.2: Prostredie kolaboratívneho editora Figma

5

Návrh modelu

...

6

Implementácia

...

7

Výsledky

...

8

Záver

...

Literatúra

- [CS16] W. Chansuwath and T. Senivongse. A model-driven development of web applications using angularjs framework. In *2016 IEEE/A-CIS 15th International Conference on Computer and Information Science (ICIS)*, pages 1–6, June 2016.
- [Fou17] Node.js Foundation. Docs | node.js [online], Máj 2017.
<https://nodejs.org/en/docs/>.
- [Goo17] Inc. Google. Angularjs api docs [online], Máj 2017.
<https://docs.angularjs.org/api>.
- [KAW⁺14] M. Kuhara, N. Amano, K. Watanabe, Y. Nogami, and M. Fukushima. A peer-to-peer communication function among web browsers for web-based volunteer computing. In *2014 14th International Symposium on Communications and Information Technologies (ISCIT)*, pages 383–387, Sept 2014.
- [KNGR13] S. Kode, K. Nagaraju, L. Gollapudi, and S. K. Reddy. Using mediawiki to increase teaching expertise in engineering colleges. In *2013 IEEE Fifth International Conference on Technology for Education (t4e 2013)*, pages 204–205, Dec 2013.

- [Med17a] MediaWiki.org. Help:formatting mediawiki - [online], Máj 2017.
<https://www.mediawiki.org/wiki/Help:Formatting>.
- [Med17b] MediaWiki.org. Manual:developing extensions [online], Máj 2017.
https://www.mediawiki.org/wiki/Manual:Developing_extensions.
- [Med17c] MediaWiki.org. Mediawiki docs [online], Máj 2017.
<https://doc.wikimedia.org/mediawiki-core/master/php/>.
- [Pee17] Peerjs.com. Peerjs documentation [online], Máj 2017.
<http://peerjs.com/docs/#api>.
- [tea17] Fabric.js team. Fabricjs doc [online], Máj 2017.
<http://fabricjs.com/docs>.
- [VSB16] S. Vashishth, Y. Sinha, and K. H. Babu. Addressing challenges in browser based p2p content sharing framework using webrtc. In *2016 IEEE 30th International Conference on Advanced Information Networking and Applications (AINA)*, pages 850–857, March 2016.
- [W3S17] W3Schools.com. Angularjs tutorial [online], Máj 2017.
<https://www.w3schools.com/angular/>.

Zoznam obrázkov

4.1	Editor SVG-edit	15
4.2	Editor Figma	16