

Prostredie pre spoluprácu viacej užívateľov

Projekt ITU, 2015Z

Číslo projektu: 79

Číslo a názov tímu: 160. Tým xvidaj00

Autor: Marek Marušic (xmarus05)

Další členové tímu: Juraj Vida (xvidaj00), Eduard Rybár (xrybar04)

Termín riešení: 21. 9. - 18. 12. 2015

Abstrakt

Tento projekt má za cieľ návrh, implementáciu a otestovanie grafického užívateľského prostredia pre aplikáciu, ktorá umožňuje spoluprácu viac užívateľov v realnom čase. Naša práca sa snaží zamerať na definíciu kľúčových momentov interakcie spolupracujúcich užívateľov. V našej práci sa zameriame na vytvorenie plochy na ktorú môžu všetci užívatelia kresliť a editovať jej objekty naraz. Budeme sa snažiť implementovať systém uzamykania objektov podobný v databázach (v momente keď začne užívateľ pracovať s vlastnosťou objektu, tá sa uzamkne a ostatní užívatelia musia čakať kým dokončí prácu). Naš projekt bude implementovaný ako webová aplikácia.

Cílové požadavky na aplikaci a její rozhraní

- Hlavným cílem je jednoduchost používání a rozložení ovládacích prvků UI a ulehčení spolupráce viacerých užívateľov v reálnom čase na rovnakom projekte.
- To či sme ciele splnili zistíme na základe feedbacku od užívateľov a pomocou testovania aplikácie
- Táto aplikácia je určená pre tímy užívateľov, ktorý spolupracujú na rôznych projektoch. Jej cieľom je zjednodušenie tejto spolupráce aj pre tímy, ktoré nemôžu byť fyzicky na jednom mieste, a preto naša zdieľaná tabuľa im umožní jednoduchšiu interakciu.
- Spokojný užívateľ sa spozná podľa rýchlosti a jednoduchosti vývoja projektov a práce na projektoch v našej aplikácii.

Studium uživatelů, UI a testování

- Cieľová skupina sú užívatelia, ktorý potrebujú jednoduchým a rýchlym spôsobom zdieľať svoju prácu, a tak isto spolupracovať na svojich prácach s ďalšími užívateľmi. Táto skupina prináša obmedzenie UI na čo najjednoduchšie ale zároveň poskytujúce dostačujúce možnosti editácie.
- Cieľová skupina sú užívatelia, ktorý majú aspoň minimálne skúsenosti s editovaním objektov na whiteboarde (malovanie vo win atd..)
- Podobné riešenia implementoval google napríklad v google drive kreslenie. Ďalšie podobné riešenie je napr. Etherpad (spolupráca na textových dokumentoch. Dokáže zvýrazňovať text rôznymi farbami, na základe autora).
- Vhodný spôsob testovania, je zameranie sa na kľúčové momenty, ktoré sú veľmi komplexné. V týchto kľúčových momentoch je potrebné otestovať rôzne kolízie a pokusy o editáciu rovnakých objektov súčasne viacerými užívateľmi.

Návrh GUI

- Hlavné funkcie aplikácie sú vytváranie nových whiteboard dokumentov kde budú užívatelia schopní pridať rôzny text, obrázky, objekty a budú mať možnosť editovať pridané objekty. Všetci užívatelia budú schopní spolupracovať na daných dokumentoch naraz v reálnom čase.
- Užívateľovi treba zobrazíť menu, v ktorom si vyberie na akom projekte chce pracovať. Taktiež toolbox, v ktorom si vyberie nástroje, ktorými bude možné upravovať alebo vytvárať objekty. Bude možné meniť veľkosť objektov, meniť ich pozíciu na tabuli, meniť ich farby, atd.
- GUI bude vytvorené ako web, pre jednoduchší prístup a prenositeľnosť (keďže každý užívateľ internetu dokáže otvoriť webovú stránku). Taktiež pre jednoduchšiu distribúciu konečným používateľom bez nutnosti inštalovať zbytočný software.
- Mockup gui by mal mať vo vrchnej časti menu pre správu dokumentov a nastavenia aplikácie a pre zobrazenie informácií o užívateľovi. V ľavej časti dokumentu by sa mal nachádzať toolbox kde budú umiestnené nástroje na úpravu a

vytváranie objektov. V strede sa bude nachádzať pracovná plocha kde budú užívatelia pracovať.

Návrh testování

- Budeme sa snažiť otestovať GUI na skupinách užívateľov rôznych vekových kategórií, ktorý sa vedia pripojiť na internet.
- Testovať sa bude funkcionálna v prípadoch spolupráce na rovnakých objektoch a ich vlastnostiach.
- Testery budú mať za úlohu vytvoriť, zdieľať a spolupracovať na projektoch.
- Vytvárať objekty rozneho druhu, a roznych vlastností.
- Pocas alfa testovania budú testery developery aplikácie a budú sa snažiť odchytiť čo najviac chýb, ktoré sa môžu objaviť počas testovania.
-

Studium realizace GUI

- Pre riešenie aplikácie je vhodné použiť programovacie jazyky Java a jej nástroje pre vyvíjanie GUI kvoli jej dobrej prenositeľnosti. Ďalej by sme mohli pre aplikáciu použiť C/C++ v kombinácii s Qt kvoli rýchlosti C++.
- My sme pre implementáciu vybrali webové technológie HTML5, CSS, Javascript, jquery v kombinácii s node.js. Webové technológie sme vybrali z dôvodu prenositeľnosti, bez potreby inštalácie nových aplikácií kvoli podpore. Každému užívateľovi stačí obyčajný webový prehliadač. Toto je výhoda, kvôli rozmanitosti cieľovej skupine užívateľov.
- Pre spoluprácu užívateľov použijeme technológiu Operational transformation, ktorá zrýchli zobrazovanie zmien v editovaných dokumentoch počas spolupráce viacerých užívateľov v rovnakom čase.
- Pre real-time zobrazenie editovania a pridávania objektov sme použili framework socket.io, ktorý pracuje s HTML5 socketami. HTML5 sockety fungujú na princípe pripojenia klient-server. Toto nám umožní jednoducho vyslať informácie o editácií do „rooms“ [1]. Každý projekt tvorí jednu room kde sú pripojení užívatelia a týmto užívateľom sa posielajú dáta z tejto room.

Návrh/popis back-endu

- Pre prácu s node.js používame express.js¹ framework, ktorý umožní rýchlejší a jednoduchší vývoj aplikácie. Tento framework používa MVC (Model-view-controller) architektúru, ktorá nám umožní jednoduché prepojenie GUI s backend-om aplikácie.

- Mysql databázu použijeme pre uloženie informácií o užívateľoch a vytvorené projekty.
- Pre jednoduchší vývoj GUI použijeme css framework Bootstrap a google material icons², ktoré ponúkajú jednoduché použitie ikon pre jednotlivé tlačítka.

Programování back-endu

- V back-ende je použitý Socket.io – knižnica pre real-time komunikáciu server/client a express.js framework.
- Editáciu objektov a prácu s nimi zabezpečuje paper.js, knižnica pre HTML5 canvas napísaná v javascripte[2].
- Informácie o pripojenom užívateľovi uchováame v session pomocou express-sessions knižnicou a ukladáme ich do socketov.
- Pre rýchle posielanie údajov sme použili HTML5 websockety. Prácu s nimi zabezpečuje súbor *app.js*, ktorý obsahuje serverovú časť obsluhy socketov a predstavuje samotný server.
- Vytvorili sme funkcie ako napr. *drawLine()*, *drawCircle()* atď, jednotlivé nástroje pre editáciu *dragTool()*, *circleTool()*, *lineTool()*, *curveTool()* atď.

Programování GUI

- GUI sme vytvorili pomocou HTML5, CSS a jade templátovací stroj.
- Spojenie GUI sme realizovali pomocou pridania jquery eventov jednotlivým tlačítkam a do týchto eventov sa pridali volania potrebných backend a frontend funkcií.
- Jednotlivé templaty, ktoré reprezentujú html výzor boli implementované pomocou jade, do týchto templatov sa z backendu posielajú dáta pomocou routera z express.js. Tento router zabezpečuje rendrovanie a spracovanie dotazov.
- HTML elementy boli naštýlované pomocou bootstrap.css a oživené pomocou jquery.

Testování

- Testovanie prebiehalo na skupinách užívateľov po 3 ľuďoch naraz, keďže je zatiaľ možné pripojenie iba 3 užívateľov v rovnakom čase. Užívatelia boli rozmiestnení každý do inej miestnosti.
- V prvom kroku im bolo prikázané vytvoriť si vlastný projekt a skúšať jednotlivé nástroje v danom projekte.
- Následne sme im prikázali aby sa všetci naraz pripojili do predpriraveného projektu, v ktorom boli pripojení všetci užívatelia a mali 5 minút náhodne skúšali

kresliť rôzne objekty naraz v rovnakom projekte aby sme videli ako funguje ich kolaborácia.

- Ďalších 5 minút mali za úlohu sa pokúšať upravovať (posúvať, meniť veľkosť, meniť tvary atď) objekty bez ohľadu na to či dané objekty už niekto v rovnakom čase upravoval. Týmto sme odskúšali či je správanie aplikácie korektné počas pokusu o detiáciu rôznych objektov v rovnakom čase.
- Počas celého testovania mali užívatelia raz za čas napísať nejaký text do chatu. Taktiež ho mohli použiť ako konverzáciu medzi sebou.
- Takýchto skupín sme použili 6(kedže sme mali k dispozícií len kamarátov a rodinných príslušníkov).
- Na konci testov sme sa vždy spýtali ako sa im páčilo pracovať v našej aplikácii.

Výsledky a závä

- Po prvom alfa testovaní sme zistili že užívatelia by veľmi radi komunikovali medzi sebou, takže sme sa rozhodli naimplementovať jednoduchý chat.
- Z nameraných výsledkov sme usúdili že GUI je dostatočne jednoduché čo bol náš zámer a ľudia boli spokojní s používaním našej aplikácie.
- Veľa nových vecí sa nám podarilo získať po konverzácií s testovacími užívatel'ami.

Týmová spolupráce

- Počas týmovej spolupráce sme sa naučili rozdeliť si primerane úlohy a ako lepšie spolupracovať. Naučili sme sa dopĺňať jeden druhého, keď sme konzultovali postupy implementácie a návrhu.
- Pomohli sme jeden druhého naučiť používať nové, pre nás neprebádané technológie.

Závä

- Táto aplikácia splnila prvotné predpoklady a ciele.
- Rozhodli sme sa po odovzdaní projektu pokračovať v rozvíjaní a zdokonalovaní tejto aplikácie, keďže sa nám zapáčila použitá technológia a boli sme prekvapení jej obrovskými možnosťami a samozrejme je veľkou rýchlosťou.

Reference

- [1] *Socket.io Rooms and Namespaces* [online]. [cit. 2015-12-13]. Dostupné z:
<http://socket.io/docs/rooms-and-namespaces>
- [2] *Paper.js Reference* [online]. [cit. 2015-12-13]. Dostupné z: <http://paperjs.org/reference/>