

Demonstrace křivek v počítačové grafice

Projekt ITU, 2016Z

Číslo projektu: 9

Číslo a název týmu: 136. Tým xtisov00

Autor: Peter Tisovčík (xtisov00)

Další členové týmu: Klára Nečasová (xnecas24)

Termín řešení: 19. 9. – 14. 12. 2016

Abstrakt

Cieľom tohto projektu je vytvoriť program, ktorý bude demonštrovať vlastnosti kriviek a vzdelávať pútavou formou.

Užívateľské rozhranie je potrebné zamerať na prehľadnosť jednotlivých ovládacích prvkov a jednoduchosť použitia pri manipulácii s nimi.

Možné problémy, ktoré je potreba riešiť, sú konzistencia dát pri väčšom množstve príkladov a vlastností.

Motiváciou je vytvorenie jednoduchej aplikácie, ktorá graficky demonštruje vlastnosti jednotlivých kriviek s jasným popisom jednotlivých vlastností. Užívateľom priblihuje prácu s krivkami zábavným a jednoduchým spôsobom.

Cílové požadavky na aplikaci a její rozhraní

Cieľom je vytvoriť aplikáciu, ktorá bude demonštrovať rozdieli medzi jednotlivými krivkami. Rozdieli sa budú demonštrovať vizuálne v podobe zobrazovania jednotlivých kriviek, ktorým bude možné meniť vlastnosti a tak sledovať ich zmeny. Výsledná aplikácia by mala umožniť pochopiť rozdieli medzi jednotlivými krivkami a medzi nastavením jednotlivých vlastností krivky.

Aplikácia by mala byť jednoduchá, ale zároveň intuitívna pre nastavovanie jednotlivých vlastností. Určite by v nej nemali chýbať príklady, ktoré názorne demonštrujú rozdiely a popisy jednotlivých prvkov a vlastností.

Aplikácia by mala slúžiť pre študentov na zopakovanie vedomostí o krivkách, ale aj pre učiteľov na demonštráciu vlastností kriviek.

Mala by umožňovať pridávať ďalšie príklady kriviek a ich ukladanie, prípadne načítanie. Ďalej by mala umožňovať dostatočné zvýraznenie riadiacich bodov a uzlových vektorov.

Po zapnutí aplikácie by sa mal užívateľ jednoducho zorientovať v ovládaní, nastavovaní vlastností, vytváraní, pridávaní a ukladaní kriviek.

Studium cílové skupiny a případy použití

Cieľová skupina užívateľov je vzhľadom na povahu aplikácie veľmi špecifická. Aplikácia sa zameriava na konkrétnu oblasť, preto sa predpokladá určitá znalosť základných princípov a jednotlivých vlastností kriviek.

Do cieľovej skupiny patria učelia a študenti. Aplikáciu je vhodné používať ako doplnok pri vysvetľovaní vlastností kriviek alebo pre študentov, ktorí si chcú overiť, poprípadne zopakovať svoje vedomosti o krivkách. Je potrebné mať na pamäti, že aj osoby technicky nie zdatné musia pochopiť účel a považovať ovládanie za intuitívne a jasné.

Kompromis medzi účelovosťou a grafickým dizajnom treba smerovať cestou prioritného zobrazenia kriviek, ktoré je príjemné pre študovanie.

Existující řešení

Existujúce riešenie podobnej problematiky Ferguson Curve applet¹. Aplikácia umožňuje nastavovanie polohy dvoch koncových bodov a orientácie tečných vektorov. Pod vykreslenou krivkou je stručne a výstižne vysvetlené ovládanie. Koncový bod, prípadne šípka tečného vektoru sa zvýrazní pokiaľ na ňu presunieme kurzor myši.

Bohužiaľ toto zvýraznenie by malo byť výraznejšie. Veľmi nevhodný je zvolený spôsob prechodu do pôvodného stavu, ktorý je realizovaný kliknutím mimo zobrazenú krivku.

Žiadúcim cieľovým efektom je, aby boli zobrazované prvky dostatočne viditeľné a aby bola práca s myšou jednoduchá.

¹<http://www2.mat.dtu.dk/info/mathematics/VIDIGEO/applets/book5.html>

Ďalšou podobnou aplikáciou je NURBS Demo – Evaluator for Non Uniform Rational B-Spline². Jedná sa o aplikáciu, ktorá ma praktickým spôsobom pomôcť k pochopeniu NURBS krivkám.

Má prehľadne spracované rozhranie a rozloženie prvkov. Pre začiatočníkov je veľmi vhodná možnosť zobrazenia už preddefinovaných kriviek. I keď je aplikácia pomerne zložitá, tak k rýchlemu pochopeniu jednotlivých položiek napomáhajú informatívne vysvetlivky. Pri zmene typu krivky zostávajú možnosti, ktoré platia len pre NURBS krivky, čo beriem ako veľkú nevýhodu pre začiatočníkov.

Z tejto aplikácie vyplýva, že je potrebné si dať pozor pri zobrazovaní nastavení pri rôznych typoch kriviek. Taktiež je dobré, čo najlepšie popísať jednotlivé vlastnosti pre presné pochopenie danej vlastnosti.

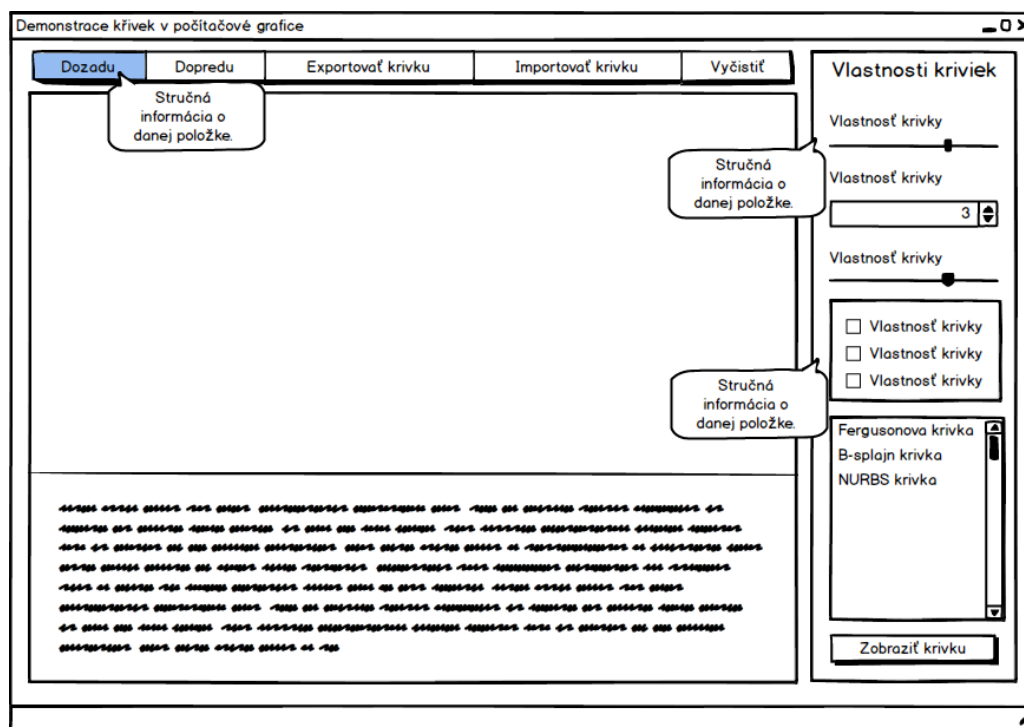
Návrh GUI

Po zapnutí sa v aplikácii zobrazí jednoduchý pomocník, ktorý užívateľovi popíše základné ovládanie myšou alebo klávesnicou. Po zoznámení sa zobrazí užívateľovi program, ktorý bude vertikálne rozdelený na dve časti. V ľavej časti sa bude nachádzať vykresľovacie plátno, kde sa budú vykresľovať krivky. V pravej časti sa bude nachádzať ovládacia časť. Po prejdení myšou nad ovládacie alebo nastavovacie element sa zobrazí krátky popis daného elementu.

V pravej časti sa budú zobrazovať vlastnosti kriviek. Pod týmito nastaveniami bude výber krivky, z rolovacieho zoznamu, ktorá sa má zobraziť. Krivky budú usporiadané hierarchicky. Budú zoskupené podľa typu krivky, takže sa bude môcť jednoducho vyhľadávať krivka určitého typu. Nad nimi sa budú zobrazovať jednotlivé vlastnosti v štýle posuvníka, boxu, zaškrtávacieho boxu alebo boxu, ktorý bude obsahovať predpripravené hodnoty. Zmenené nastavenie krivky sa prejaví okamžite bez nutnosti ukladania.

V ľavej časti sa bude nachádzať zobrazovacia časť, ktorá bude rozdelená na dve horizontálne časti, nad ktorými sa budú nachádzať ovládacie prvky. Medzi ovládacie prvky bude patriť krok späť, krok vpred, ukladanie, načítanie nastavení a vyčistenie pracovného priestoru. Vrchná časť bude slúžiť pre zobrazenie krivky, bodov a vektorov. V druhej časti sa budú zobrazovať informácie o jednotlivých vlastnostiach krivky a prípadné typy na správne nastavenie krivky.

Pre užívateľov, ktorí budú aplikáciu používať častejšie je vhodné pripraviť klávesové skratky, ktorými bude môcť realizovať tie isté operácie ako myšou.

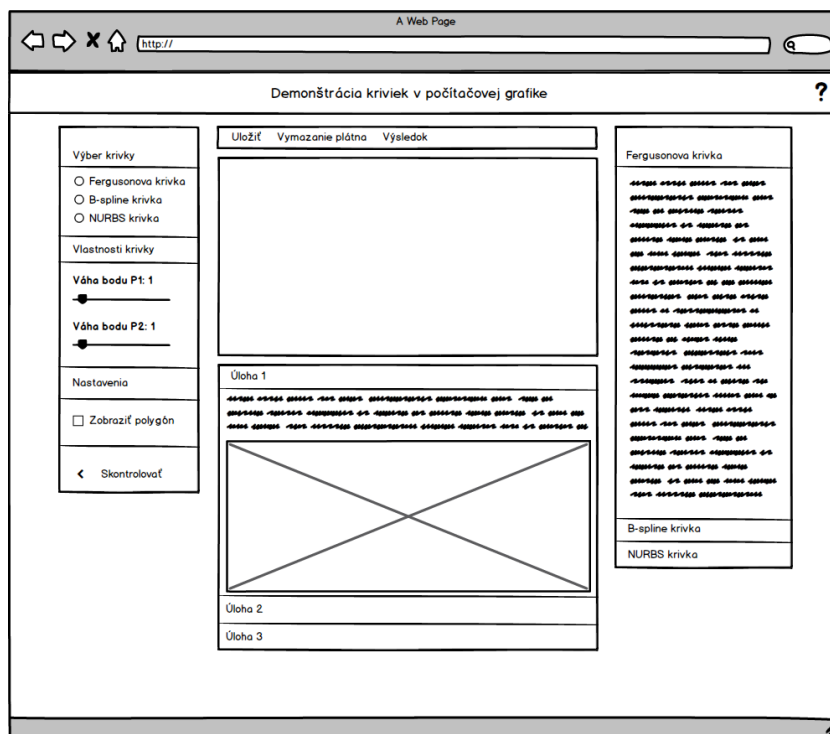


Obrázok 1: Mockup aplikácie

²<http://nurbscalculator.in/>

Spoločný návrh GUI

Po konzultácií s druhým členom tímu sme dospeli k spoločnému návrhu GUI a jeho funkcionalite. Po načítaní webovej stránky sa zobrazí stránka rozdelená do niekoľkých panelov. Vo vrchnej časti sa nachádza nápoveda a názov aplikácie. Pod touto časťou je stránka rozdelená do troch častí. V ľavej časti sa nachádza zoznam kriviek, ktoré je možné využiť k vypracovaniu úlohy. Pod zoznamom kriviek sa nachádzajú jej vlastnosti a tlačidlá na pridanie alebo odstránenie bodu. V spodnej časti je možné zobrazit' alebo skryť riadiaci polygón a skontrolovať vypracovanú úlohu. V strednej časti stránky sa nachádza malé menu, ktoré obsahuje operácie, ktoré umožňuje pracovať s kresliacim plátnom. Pod kresliacim plátnom sa nachádza zoznam úloh. Prednastavená je úloha 1. Po kliknutí na inú úlohu sa aktuálne zobrazená úloha zavrie a otvorí sa požadovaná úloha. V každej úlohe je stručný popis a obrázok ako má vypadáť výsledok. V pravej časti sa zobrazujú informácie o vybraných krivkách. Po kliknutí na určitý typ krivky sa k nej otvorí príslušné kresliace plátno, úlohy (stredná časť) a popis krivky (pravá časť). Výsledok vypracovanej úlohy sa zobrazí v menu.



Obrázok 2: Spoločný mockup aplikácie

Návrh back-endu

V prípade desktopovej aplikácie je potrebné si vybrať operačné systémy, pre ktoré by bola aplikácia určená. V prípade multiplatformovej aplikácie je potrebné zvážiť, či by nebolo vhodné vytvoriť webovú aplikáciu, pri ktorej nie je potrebné riešiť kompatibilitu na úrovni operačných systémov ale na úrovni webových prehliadačov.

V prípade webovej aplikácie je možné použiť JavaScript a HTML5 (canvas), ktoré je priamo určené na kreslenie a spojením spolu s JavaScript-om by sa dalo docieľiť vykresľovanie v rámci kresliaceho plátna a zmena pozície bodov pomocou myši. Problém by bol s ukladaním vytvorených kriviek. Ich zdieľanie by sa dalo docieľiť ukladaním ako obrázok. Použitý by bol framework Bootstrap³ pre vizuálnu časť. JavaScript-ová knižnica ibiblio⁴ bola použitá pre vykreslenie B-spline a NURBS krivky. Pre vykreslenie Fergusonovej krivky by bol použitý Canvas Bézier Curve⁵.

Opačný prípad je desktopová aplikácia, kde je potrebné riešiť multiplatformovosť ak sa nejedná o Javu, kde by mohli aj tak nastať problémy pri práci s nejakou špecifickou knižnicou. Prípadne využiť knižnice, ktoré sú dostupné na rôzne platformy. Vhodným programovacím jazykom by bolo C++ a voľne dostupné knižnice GSL⁶ alebo NURBS++⁷. Pre GUI by bolo vhodné využiť Qt framework⁸.

³<http://getbootstrap.com/>

⁴<http://www.ibiblio.org/>

⁵<http://blogs.sitepointstatic.com/examples/tech/canvas-curves/bezier-curve.html>

⁶<https://www.gnu.org/software/gsl/>

⁷<http://libnurbs.sourceforge.net/>

⁸<https://www.qt.io/>

Implementace back-endu

Pre výslednú implementáciu bola zvolená webová aplikácia, ktorá bola postavená na JavaScript-ových knižniciach. Použité knižnice bolo treba upraviť aby vyhovovali nášmu návrhu. Bolo nutné doprogramovať niekoľko dôležitých funkcií, ktoré sme navrhli. Medzi najdôležitejšiu funkčnosť aplikácie považujeme kontrolu nakreslenej krivky zo vzorovou, ktorá sa vykonáva na základe zadaného požadovaného vstupu a výstupu, ktorý sa zobrazuje v kresliacom plátne. Ďalej bolo treba doplniť zmenu váh jednotlivých vektorov, reset plátna a exportovanie obrázka. Kontrola správnosti nakreslenej krivky sa vykonáva porovnaním bodov v kresliacom plátne s bodmi definovanými v popise úlohy, ktoré je možné vidieť len v zdrojovom kóde. Reset plátna je vlastne len znovu načítanie bodov na preddefinovaných súradniciach. Pre export obrázka je použitý JavaScript, ktorý vygeneruje obrázok, prípadne odkaz na jeho stiahnutie. Tieto jednotlivé funkcie bolo nutné implementovať pre jednotlivé typy kriviek samostatne. Aplikácia obsahuje možnosť pridávať a odstraňovať priamo do kresliaceho plátna.

Návrh užívateľských testů

Testovanie aplikácie má za cieľ preukázať validnosť aplikácie, teda je potreba pred sprístupnením aplikácie nájsť chyby v aplikácii. Testovanie funkčnosti je potreba integrovať už pri návrhu aplikácie, čo značne zjednodušuje odhalovanie chýb a vo finálnej aplikácii sa prejavujú vo veľkej miere iba chyby zanesené integrovaním a nasadením. Vhodné testovanie pre našu aplikáciu bude na zvolenej množine študentov s rôznou znalosťou práce s počítačom. Vhodným spôsobom testovania je testovanie vo fázach, kde sa náročnosť úloh nemení a mení sa len testovacia skupina a prostredie, v ktorom sa testuje. Medzi ďalšie možnosti testovania patrí sledovanie pádov aplikácie a ich oznámovanie.

Testovanie by prebiehalo na skupine ľudí okolo 20-50 rokov. Medzi testovaciu skupinu budú patriť študenti. Môže sa predpokladať, že užívatelia budú mať rôzne skúsenosti s počítačmi. Testovanie bude prebiehať v niekoľkých fázach a na užívateľov bude dohliadať expert, ktorý im prípadne pomôže a poradí.

V prvej fáze sa bude testovať medzi vývojármi, aby sa odladili závažné chyby a doladili detaily aplikácie. Testovanie bude prebiehať testovaním funkčnosti ovládacích prvkov a funkcionality aplikácie ako takej. Cieľom testovania je odhaliť problémové časti programu a rozhrania s pohľadom vývojárov.

V druhej fáze sa bude testovať na troch užívateľoch. Testovať sa bude v uzavretej skupine študentov. Testovanie bude prebiehať pod dohľadom vývojárov a prípadné problémy sa budú rovnako konzultovať s vývojármi. Na testovanie budú pripravené testovacie protokoly a čas na ich vypracovanie nebude limitovaný. Cieľom testovania je odhaliť problémové časti grafického užívateľského rozhrania.

V tretej fáze sa bude testovať v počte päť študentov. Testovať sa bude na školských počítačoch a na testovanie bude opäť pripravený testovací protokol. Čas testovania bude obmedzený na 15 minút. Cieľom testovania bude získať informácie z nasadenia a používania na školských počítačoch a v školskom prostredí.

V štvrtej fáze sa bude testovať v menšom počte študentov približne 5 v domácich podmienkach. Na testovanie bude pripravený testovací protokol a testovanie nebude časovo obmedzené. Cieľom testovania bude získať skúsenosti z nasadenia programu u študentov na ich počítačoch bez dohľadu experta.

V rámci testov budú poskytnuté testovacie protokoly, z ktorých sa vyhodnotí rozloženie prvkov a zložitosť ovládania. Medzi jednotlivými fázami testovania sa zapracujú prípadné pripomienky zo strany užívateľov a testovacia fáza sa bude opakovať na rovnakej skupine ľudí a bude sa skúmať, či sa podarilo naplniť očakávanie užívateľov a aplikáciu zlepšiť. Ďalej bude možné zistiť ako sa vyvíjalo hodnotenie aplikácie užívateľmi v procese testovania. Zároveň sa vyhodnotí aplikácia z funkčného hľadiska.

Nástroje pro tvorbu GUI

Pre implementáciu nášho projektu sme zvolili webové technológie. Vybrali sme Bootstrap framework, HTML5, CSS a JavaScript na vizuálnu časť stránky a z časti pre jeho back-end. Patrí medzi najpoužívanejšie frameworky. Bootstrap obsahuje sadu predpripravených komponent pre prácu s elementami na stránke. Podporuje responzívne zobrazovanie a umožňuje jednoduchú rozšriteľnosť.

Webové technológie sú nezávislé na použítom zariadení a operačnom systéme, na ktorom je aplikácia spustená. Nie je nutné riešiť inštaláciu, distribúciu aktualizácií, prípadne vytváranie rôznych verzií pre rôzne operačné systémy. Systém je možné spravovať z jedného miesta a užívateľom len sprístupniť miesto, kde sa im bude daná aplikácia zobrazovať. Nedostupnosť miesta, kde je daná aplikácia zobrazovaná, znemožňuje užívateľom ju využiť, čo sa radí k nevýhodám. Obmedzujúcim faktorom môže byť aj nutnosť pristupovať k internetu pri používaní aplikácie.

Programování GUI

Najskôr bolo potrebné premyslieť vytvorenie kostry šablóny podľa predpripraveného návrhu v Bootstrap, HTML a CSS. Následne sme implementovali kreslenie jednotlivých kriviek, Fergusonovu, B-spline a nakoniec NURBS krivku. Po implementovaní kreslenia bolo nutné dopracovať porovnávanie nakreslených kriviek zo správnym riešením. Pred koncom sme dopracovali nastavovanie váh, dynamické zobrazovanie panelov a export nakreslenej krivky ako obrázok. Všetko bolo napísané v JavaScripte. Poslednou časťou bolo doladenie farieb, rozloženia a celkového vzhľadu aplikácie.

Užívateľské testy

Testovanie aplikácie prebehlo podľa návrhu. Niektoré fázy testovania prebehli počas vývoja aplikácie, čím bolo možné podchytiť niektoré problémy hneď na začiatku.

V niektorých fázach testovania dostala osoba krátke oboznámenie z aplikáciou, jej ovládaním a významom. Na záver testovania užívateľa vyplnili krátky dotazník.

Z dotazníkom sme zistili, že aplikácia by mala byť v tmavých farbách aby neľahalo oči pri práci. Nápoveda k programu a jednotlivým krivkám by mala byť oddelená a ideálne dobre dostupná. Zvýraznenie správnosti porovnania by malo byť prehľadnejšie.

Výsledky a záver

V prvej fáze testovania medzi vývojarmi sme objavili niekoľko závažných chýb. Patrili medzi ne nesprávne umiestnené popisy, nepresnosti popisov, nepresnosti pri vykresľovaní.

V druhej fáze testovania aplikácie sme zistili, že užívateľom robí najväčší problém odhadnúť sklon a tvar krivky podľa priloženého vzoru a niekedy im bolo potrebné dopomôcť k správnejmu výsledku. K rozmiestneniu prvkov užívateľa nemali výhrady. Výhrady užívateľov padli len na príliš svetlé farby z čoho vyplývalo namáhanie očí. Toto sme zobrali do úvahy a vytvorili sme tmavú verziu aplikácie. Ďalej sme zistili, že užívateľa by privítala možnosť pridávania a odstraňovania bodov.

V tretej fáze testovania aplikácie sme zistili, že zhotovené testy boli príliš ľahké a odstránili sme riadiace polygóny zo vzorových obrázkov, čo užívateľov nútilo sa zamyslieť nad tým ako musia byť rozmiestnené jednotlivé body a váhy riadiacich polygónov. Túto možnosť sme pridali aj do aplikácie pod názvom Zobrazit řídící polygon.

V poslednej fáze testovania sme zapracovali jednotlivé pripomienky od užívateľov. Uistili sme sa, že predpripravená sada úloh bola primerane náročná a či jednotlivé farby na stránke neľahajú oči pri práci.

Týmová spolupráce

Práca v tíme prebiehala bezproblémovo. Jednotlivé návrhy a implementáciu sme preberali medzi sebou z čoho nám niekedy vznikli lepšie myšlienky a nápady ako pred spoločnou konzultáciou. Ďalšou výhodou bolo rozdelenie práce medzi členmi v tíme a tak zrýchlenie vývoja aplikácie.

Na druhú stranu bolo potrebné informovať členov tímu o aktuálnom stave zadaných úloh a priebežne na nich pracovať aj keď sa to nie vždy hodilo. Niekedy zase bolo nutné čakať na iného člena tímu pokiaľ spravil svoju časť a mohlo sa to spojiť spolu.

Záver

Naším cieľom bolo vytvoriť aplikáciu, ktorá bude študentom zaujímavým a hravým spôsobom demonštrovať jednotlivé vlastnosti kriviek a prácu s nimi. Zo získaných testov je možné usúdiť, že väčšine testovaných užívateľov sa aplikácia páčila. Nápadu užívateľov sme konzultovali a snažili sa začleniť do aplikácie aby sme rozšírili jeho funkčnosť, čo najviac. Počas jednotlivých fáz testovania sme aplikáciu vylepšovali a ďalším testovaním by sa aplikácia dala určite ešte zlepšovať.

Návrh testovacího protokolu

Úlohy

- Zobrazit' nápovedu
- Postupne načítat' úlohy k Fergusonovej, B-spline a NURBS krivke
- Načítat' Fergusonovu krivku
- Nastaviť orientáciu tečných vektorov
- Načítat' NURBS0 krivku
- Vymazať kresliace plátno
- Nastaviť stupeň a pridať riadiaci bod
- Posunúť a odstrániť riadiaci bod
- Načítat' B-splajn krivku
- Pridať riadiaci bod

1. Aké máte skúsenosti s podobnými aplikáciami?

- Žiadne
- Už som pracoval s podobnou aplikáciou

2. Ako hodnotíte rozloženie prvkov v aplikácii?

- Neprehľadné
- Primerane prehľadné
- Prehľadné

3. Ako hodnotíte vloženie nápovedy?

- Neprehľadná
- Primerane prehľadná
- Prehľadná

4. Ako hodnotíte obtiažnosť úloh?

- Ľahké
- Primerane náročné
- Náročné

5. Ako hodnotíte prehľadnosť vykreslených kriviek?

- Neprehľadné
- Primerane prehľadné
- Prehľadné

6. Ako hodnotíte prehľadnosť nastavení kriviek?

- Neprehľadná

- Primerane prehľadná
- Prehľadná

7. Vyhovovalo Vám ovládanie aplikácie?

- Áno
- Nie (Uveďte dôvod)

8. Ktorá uloha alebo jej časť bola najt'azšia?

9. Čo by ste v aplikácii privítali prípadne zmenili?