

KATALÓG POŽIADAVIEK

IMS SPEKTROMETER

Bencz Vladimir

Krivánek Eduard

Kočalka Andrej

Harnádek Juraj

Obsah

1.	Katalóg požiadaviek.....	6
1.1.	Účel.....	6
1.2.	Využitie systému.....	6
1.3.	Slovník pojmov	6
1.4.	Odkazy a referencie.....	6
1.5.	Prehľad nasledujúcich kapitol	6
2.	VŠEOBECNÝ POPIS.....	7
2.1.	Perspektíva systému.....	7
2.2.	Funkcie systému	7
2.3.	Charakteristika používateľa	7
2.4.	Všeobecné obmedzenia	7
2.5.	Priebeh merania	7
3.	ŠPECIFICKÉ POŽIADAVKY	8
3.1.	Funkčné požiadavky	8
3.2.	Požiadavky rozhrania.....	12
4.	Návrh	12
4.1	Účel	12
4.2	Rozsah.....	12
4.3	Definície a skratky	12
5.	Špecifikácia vonkajších interfejsov	13
5.1.	Spektrometer	13
5.2.	ShieldBuddy TC275.....	13
5.3.	Meranie iónovej mobility	13
6.	Dátový model	13
6.1.	Formáty súborov	13
6.2.	Komunikačné protokoly	13
7.	Návrh používateľského rozhrania.....	14
7.1.	Názov projektu	14
7.2.	Nastavenia aplikácie.....	14
7.3.	Nastavenie Mobility	15
7.4.	Sekcia grafu	16
8.	Návrh implementácie	18
8.1.	Class diagram.....	18
8.2.	Stavový diagram - Hlavný graf	18

8.3.	Diagram komponentov.....	19
9.	Testovacie scenáre	20
9.1.	Test pripojenia prístroja a nastavenia portu	20
9.2.	Test nového merania.....	20
9.3.	Test zmeny nastavení	20
9.4.	Test uloženia nastavení	20
9.5.	Test načítania nastavení.....	21
9.6.	Test uloženia hlavného grafu	21
9.7.	Test možností generovania intenzitného grafu.....	21
9.8.	Test generovania intenzitného grafu z hlavného grafu.....	21
9.9.	Test generovania intenzitného grafu zo súboru.....	21
9.10.	Test uloženia intenzitného grafu.....	21
9.11.	Test zmeny farieb intenzity	21
9.12.	Test zobrazenia zrkadlového grafu.....	22
9.13.	Test aplikovania mobility.....	22

1. Katalóg požiadaviek

1.1. Účel

Toto je katalóg požiadaviek k systému na obsluhu IMS Spektrometra, ktorý vznikol ako projekt na predmet 'Tvorba informačných systémov' na fakulte matematiky fyziky a informatiky UK v Bratislave v odbore aplikovaná informatika. Je určený zadávateľom, užívateľom a komukoľvek kto bude so systémom pracovať, alebo sa chce o ňom niečo dozvedieť. Tento dokument vznikol ako analýza informácií ktoré boli získané z rozhovoru so zadávateľom projektu a bol spísaný vývojármi softvéru. Dokument je záväzný pre obe strany, pričom je dôležitá hlavne kapitola 3. ktorá obsahuje kompletný zoznam požiadaviek ktoré by mal systém spĺňať.

1.2. Využitie systému

Cieľom projektu je vytvoriť systém pre grafickú reprezentáciu nameraných údajov Spektrometrom a sprehľadniť získané údaje. Produktom je informačný systém pre katedru experimentálnej fyziky, ktorý umožňuje vizuálne zobrazenie aktuálne meraných hodnôt z pripojeného senzora (shield buddy) a ich ukladanie ako CSV súbor. Systém zabezpečí digitalizáciu hodnôt zo spektrometra, ich spracovanie a ich zobrazenie. Systém slúži na monitorovanie zrážania iónov vo spektrometri a umožňuje jeho ovládanie, ako dĺžku otvorenia brány pre zrážanie iónov.

1.3. Slovník pojmov

- **CSV** – hodnoty oddelené čiarkou
- **Digitalizácia** – prevod informácií z analógového do digitálneho tvaru
- **Hlavný graf** – graf na ktorom sa zobrazia agregované dáta v reálnom čase (bod 3.1.3)
- **Intenzitný graf** – je 3D reprezentácia (2D+farba) viacerých agregovaných dát (3.1.12)
- **Hlavný intenzitný graf** – Intenzitný graf tvorený z hlavného grafu v reálnom čase (3.1.13)
- **Cyklus merania** – cyklus za ktorý mikrokontrolér pošle namerané údaje (bod 3.1.26)
- **Agregované dáta** – vektor obsahujúci spriemerované údaje z cyklov merania po zložkách za dobu definovanú používateľom
- **Shield buddy TC 275** – multiprocesorový mikrokontrolér (bod 1.4)
- **Mobilita** – aplikovanie vzorca pre transformáciu X-ovej osi (bod 3.1.10)

1.4. Odkazy a referencie

- Manuál pre Shield Buddy TC 275 <https://docs-emea.rs-online.com/webdocs/159d/0900766b8159d23a.pdf>

1.5. Prehľad nasledujúcich kapitol

V 2. kapitole sa nachádza popis plánovaného systému prirodzeným jazykom, plynulými vetami bez toho, aby sme išli do veľkých podrobností. 3. kapitola obsahuje kompletný zoznam všetkých požiadaviek na systém.

2. VŠEOBECNÝ POPIS

2.1. Perspektíva systému

Systém bude slúžiť hlavne na vykreslenie hlavného grafu na ktorom sa nám zobrazia agregované dáta z prebiehajúcich cyklov meraní v reálnom čase. Systém umožňuje tieto agregované dáta ukladať do CSV súborov, alebo z nich vygenerovať intenzitný graf pre pozorovanie intenzity bodov v čase. Intenzitný graf môžeme načítať, uložiť a upravovať.

2.2. Funkcie systému

Aplikácia sa cez sériový port bude pripájať ku shield buddy, cez ktorý bude získavať údaje za jeden cyklus merania. Tieto údaje sa agregujú a vykreslia na hlavnom grafe v reálnom čase. Užívateľ bude mať možnosť si uložiť tieto údaje do CSV súboru, alebo z nich vytvárať intenzitný graf, taktiež v reálnom čase.

Každý riadok v Intenzitnom grafe zodpovedá jednému zobrazeniu v Hlavnom grafe. Kliknutím na niektorý riadok Intenzitného grafu sa dá spätne zrekonštruovať zodpovedajúci priebeh v takom tvare, ako bol zobrazený v Hlavnom grafe, v aplikácii sa označuje ako Zrkadlový graf. Užívateľ môže načítať viacero intenzitných grafov pre porovnávanie jedného s druhým.

Okrem súborov s uloženými nastaveniami aplikácia vytvára dva typy dátových CSV súborov. Prvý zodpovedá jednému zobrazeniu v Hlavnom grafe, obsahuje dva stĺpce: prvý stĺpec sú xové (časové) hodnoty, druhý stĺpec sú zodpovedajúce namerané intenzity.

Druhý typ súboru tvorí postupnosť takýchto cyklov merania, zodpovedá to Intenzitnému grafu. Obsahuje rovnaký prvý stĺpec: x-ovú, časovú hodnotu a nasleduje viacero stĺpcov, so zodpovedajúcimi intenzitami v jednotlivých cykloch meraní.

Údaje v CSV súboroch sa ukladajú vždy v netransformovanej podobe (pozri mobilita). Prvý riadok v každom type ukladaného CSV súboru obsahuje komentár, v ktorom sú všetky základné nastavenia, s ktorými graf vznikol (mobilita nie)

Aplikácia taktiež umožňuje aktualizovať nastavenie spektrometra pre spúšťanie iónov do trubice.

2.3. Charakteristika používateľa

Systém je určený pre študentov a profesorov na Katedre experimentálnej fyziky ktorí potrebujú používať Spektrometer. Systém neposkytuje prihlásenie, je určený pre jedného používateľa. Tento používateľ teda bude môcť využívať všetky funkcie aplikácie.

2.4. Všeobecné obmedzenia

Systém potrebuje shield buddy TC275, ktorý slúži ako komunikačný interface medzi spektrometrom a aplikáciou.

2.5. Priebeh merania

Priebeh fyzikálneho merania je približne nasledovný: v meracom prístroji na jednom mieste po vytvorení pulzu vznikajú ióny, ktoré následne letia trubicou, v ktorej je elektrické pole a to ich usmerňuje, aby doleteli až po detektor. Podľa vlastností iónu prelet trvá rôznu dobu a preto rozličné ióny detektor zachytí v rozličnom čase.

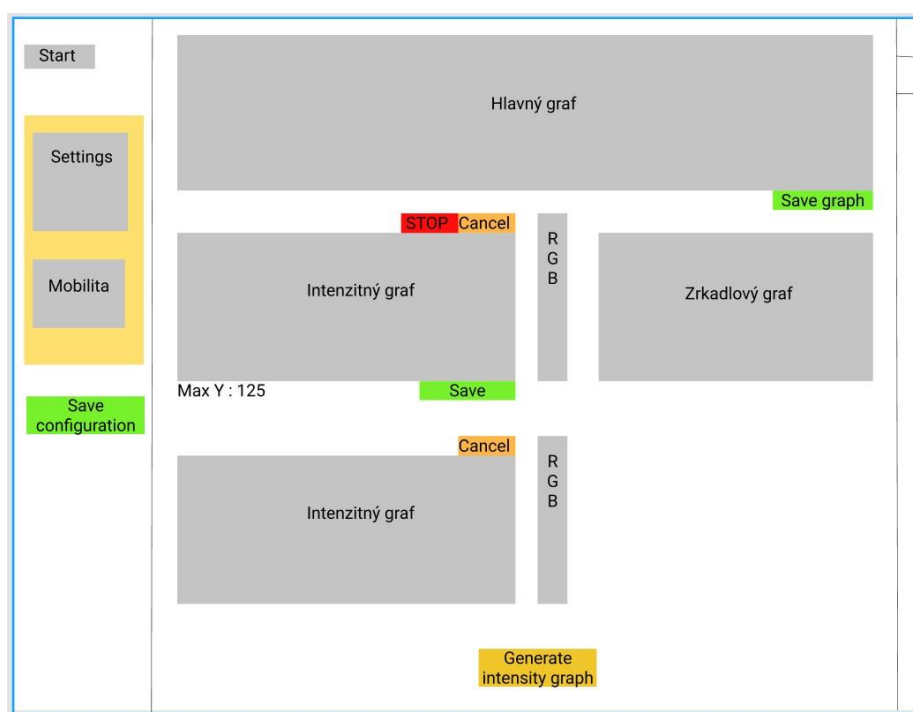
Približne po 20 ms sa už dopad ďalších iónov neočakáva a cyklus merania sa ukončuje. Počas cyklu merania sa (s nastaviteľnou frekvenciou) neustále sníma intenzita vybudenia detektora. Takéto cykly merania sa opakujú jeden za druhým.

Časovo zodpovedajúce okamihy v jednotlivých cykloch merania sa krížom cez niekoľko za sebou nasledujúcich cyklov merania priemerujú. Až takýto vektor spriemerovaných hodnôt sa zobrazuje v Hlavnom grafe.

3. ŠPECIFICKÉ POŽIADAVKY

3.1. Funkčné požiadavky

3.1.1. Nasledujúci ilustračný obrázok predstavuje predbežný dizajn aplikácie.



Obrázok 3.1.1

Na obrázku 3.1.1 môžeme vidieť

- Panel na ľavom rohu pre riadenie aplikácie
 - o „start“ button pre spustenie merania (bod 3.1.2)
 - o kolónku settings, ktorá bude konfigurovať spektrometer (bod 3.1.23)
 - o kolónku mobility, textové polia pre užívateľa s checkboxom „aplikovať“ (bod 3.1.10)
 - o tlačidlo „save configuration“, ktorá nám uloží údaje zo settings a mobility (bod 3.1.7) - V strede nám bude permanentne zobrazený hlavný graf (bod 3.1.3).
- Tlačidlo „generate intensity graph“ nám umožní generovať intenzitný graf (bod 3.1.12)
 - o ku každému intenzitnému grafu bude prislúchať škála farieb (bod 3.1.15)
 - o ku každému intenzitnému grafu bude prislúchať zrkadlo grafu (bod 3.1.17)

- 3.1.2. Meranie sa spúšťa, resp. zastavuje tým istým tlačidlom Start, resp. Stop, ktoré mení label podľa stavu merania.
- 3.1.3. Hlavný graf bude permanentne zobrazený na hlavnom okne, kde sa budú zobrazovať aktuálne agregované dáta.
- 3.1.4. Používateľ bude mať možnosť vybrať si z dvoch spôsobov agregovania údajov. Perióda agregovania bude definovaná buď 1) časovým úsekom, alebo 2) počtom cyklov merania. (bod 3.1.24).
- 3.1.5. Hlavné okno používateľského rozhrania obsahuje editovací riadok, do ktorého používateľ vpíše meno svojho projektu (reťazec písmen/číslic bez medzier). Toto meno sa pridáva pred názvy všetkých súborov, ktoré aplikácia vytvára. Ak užívateľ nezadá meno projektu, predvolené meno súborov bude „undefined“.
- 3.1.6. Po stlačení tlačidla „save graph“ sa zobrazené údaje na hlavnom grafe uložia, vo formáte CSV, názvom projektu definovaný užívateľom (bod 3.1.5) a sufixom aktuálneho timestampu na aplikáciou preddefinované miesto (Data\Aggregated_Data\).
- 3.1.7. Po stlačení tlačidla „save configuration“ sa uložia aktuálne nastavenia aplikácie spolu s mobilitou do jedného súboru, názvom projektu definovaný užívateľom (bod 3.1.5) a sufixom aktuálneho timestampu na aplikáciou preddefinované miesto (Data\Configuration\).
- 3.1.8. Po stlačení tlačidla „load configuration“ si užívateľ môže načítať uložené nastavenia a mobilitu (bod 3.1.7) naspäť do aplikácie.
- 3.1.9. Len pre hlavný graf, ak bude kurzor myši na niektorom bode v grafe, zobrazí sa vedľa kurzoru okienko s hodnotami X-ovej a Y-ovej osi.
- 3.1.10. Mobilita je parametrizovateľná funkcia, ktorá slúži na zobrazenie údajov vo všetkých troch typoch grafov v transformovaných súradniciach. Vstupom do funkcie je pôvodná x-ová hodnota (čas t) a výstupom je nová x-ová hodnota (čas), pre zobrazenie príslušného bodu krivky grafu. Jej účel je znázornenie redukovanej pohyblivosti iónov, kvôli lepšej identifikácii iónov. Je nepriamo úmerná času. Teda ióny s vysokou mobilitou dopadnú na detektor za kratší čas, a opačne tie pomalšie ióny budú mať nižšiu mobilitu. Mobilita má parametre:
- dĺžku trubice - $L(\text{cm})$ ○ tlak plynu - $p(\text{pa})$ ○ teplotu plynu - $T(\text{K})$
 - napätie na driftovej trubici - $U(\text{kV})$ ○ normálny tlak 101325 Pa - p_0
 - 293.15 Kelvinov - T_0

Mobilitu nakoniec vyrátame vzorcom: $Ko=(L^2/U*t)[(p*To)/(po*T)]$.

Checkbox „aplikovať“ v kolónke pre mobilitu bude rozhodovať či sa má mobilita aplikovať, alebo nie.

- 3.1.11. Používateľ môže vyznačením pravouhlého rámčeka pomocou myši určiť ľubovoľný menší výsek grafu, ktorý sa má priblížiť (zoom in) a zobraziť na celej ploche grafu.

Vedľa

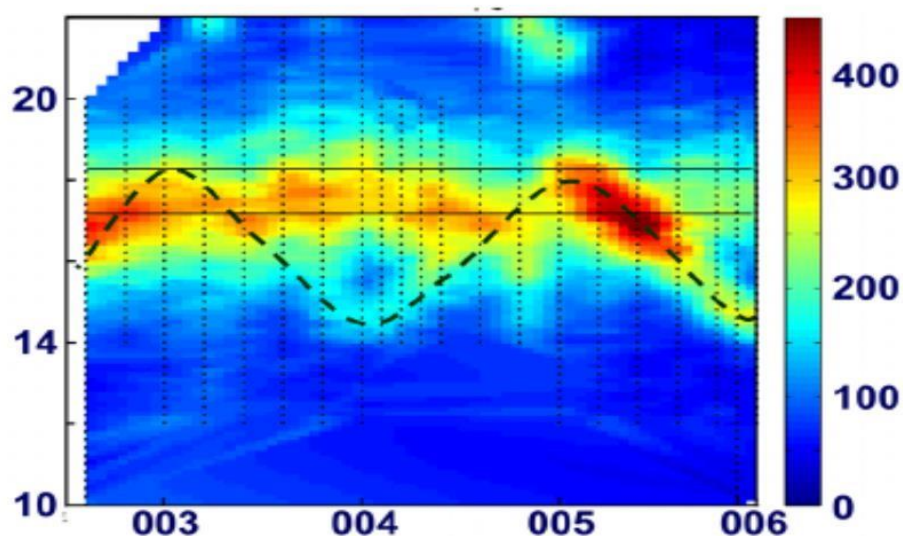
počiatku osí je malé tlačidlo, ktoré priblíženie vráti a spôsobí zobrazenie grafu v celom pôvodnom rozsahu (zoom out).

- 3.1.12. Intenzitný graf je 3D reprezentácia viacerých agregovaných dát. Na Y-ovej osi sa zobrazí poradie (počítadlo) agregovaných dát a na X-ovej časový úsek prislúchajúci k týmto dátam.

Intenzitný graf farebne rozlišuje intenzitu v bodoch, kvôli tomuto časť agregovaných dát v istých bodoch, ktoré mali vyššiu intenzitu, budú farebne rozlíšiteľné od bodov, ktoré mali nižšiu intenzitu. Farby si môže užívateľ sám nastaviť (bod 3.1.15).

Obrázok 3.1.12 znázorňuje ilustráciu intenzitného grafu.

Generovanie intenzitného grafu bude fungovať na dva spôsoby (bod 3.1.13 a 3.1.14).



obrázok 3.1.12

- 3.1.13. V prvom spôsobe zobrazenia intenzitného grafu sa užívateľovi v reálnom čase budú pripájať agregované dáta z hlavného grafu do intenzitného grafu .

- 3.1.14. V druhom spôsobe zobrazenia intenzitného grafu si užívateľ načíta už namerané hodnoty z CSV súboru.

- 3.1.15. Každý intenzitný graf bude mať svoju RGB škálu farieb, kde užívateľ môže meniť:

- 3.1.15.1. rozsah dát, ktoré majú byť zafarbené zvolenou farbou, spôsobom, že užívateľ zadá čísla do textového poľa, od - do akého rozsahu údajov sa má zvolená farba aplikovať. Úplné minimum rozsahu údajov je 0.

- 3.1.15.2. farbu, ktorá prislúcha danému rozsahu, spôsobom, že užívateľ klikne na zvolenú farbu a vyskočí mu editovacie okno farieb, kde si môže zvoliť inú farbu, alebo zadať kód farby v hexadecimálnom tvare.
- 3.1.16. Každá RGB škála farieb bude mať vlastné tlačidlo „apply to all“, ktorá zmodifikuje RGB škálu farieb (farby a aj rozsah dát) ostatným zobrazeným intenzitným grafom podľa zvoleného.
- 3.1.17. Každý intenzitný graf bude mať prislúchajúce „zrkadlo grafu“, kde sa zobrazí agregovaný graf, ktorý reprezentuje danú úsečku na intenzitnom grafe na ktorý užívateľ klikol. Pokiaľ ešte používateľ na žiaden riadok intenzitného grafu neklikol, v zrkadlovom grafe sa nič nezobrazuje.
- 3.1.18. Užívateľ bude mať možnosť pod hlavným grafom si zobraziť viaceré intenzitné grafy.
- 3.1.19. Ak množstvo intenzitných grafov presiahne obrazovku, panel, na ktorom sa grafy vykresľujú sa bude dať skrolovať.
- 3.1.20. Len pre Intenzitný graf, ktorý sa generuje z hlavného grafu bude
 - 3.1.20.1. obsahovať textfield, kde užívateľ môže zadať maximálne množstvo agregovaných dát, ktoré sa majú zobraziť na tomto intenzitnom grafe. Najstaršie údaje sa odstránia z grafu. Prednastavená kapacita bude maximálne 100 agregovaných grafov.
 - 3.1.20.2. tlačidlo „save“ uloží aktuálny stav intenzitného grafu, aj počas generovania, do jedného súboru, vo formáte CSV s názvom projektu definovaný užívateľom (bod 3.1.5) a sufix aktuálneho timestampu na aplikáciu preddefinované miesto (Data\Intensity_Data).
 - 3.1.20.3. generovanie tohto grafu sa zastavuje, resp. spúšťa tým istým tlačidlo „Start“ resp. „Stop“ ktoré mení label podľa stavu merania. Stlačením Start sa predchádzajúci zobrazený graf hlavný intenzitný graf a k nemu prislúchajúce komponenty (zrkadlový graf, rgb škála farieb) zmažú.
 - 3.1.20.4. tlačidlo „cancel“ odstráni intenzitný graf z panelu.
- 3.1.21. Oznamovanie o správnom priebehu akcie – bude to successful pop-up, ktorý oznámi užívateľovi, že jeho daná akcia prebehla úspešne (tj. Ukladanie dát, zmena správania spektrometra).
- 3.1.22. Oznamovanie o nesprávnom priebehu akcie - bude to failure pop-up, ktorý sa zobrazí s chybovou hláškou, ak sa daná akcia od užívateľa zlyhala (tj. Ukladanie dát, zmena správania spektrometra).
- 3.1.23. Nastavenia - možnosť užívateľovi dynamicky meniť nasledujúce premenné na konfiguráciu spektrometra. V zátvorkách sa určuje hranica hodnoty minima a maxima oddelenou pomlčkou:
 - 3.1.23.1. Sampling (0.5 - 50) – hustota merania v mikrosekundách
 - 3.1.23.2. Gate (1 - 20) – impulz, ktorý spúšťa ióny do driftu v spektrometri

Ale aj agregovanie cyklov merania premennými:

3.1.23.3. Repeat Time (0.1 - 60) – časová doba v sekundách na agregovanie dát

3.1.23.4. Repeat Count (1 – 3000) – počet meraní pre agregované dáta

3.1.23.5. Apply repeat time – výber režimu merania (bod 3.1.24)

Zmena nastavení spôsobí zmazanie hlavného a hlavného intenzitného grafu.

3.1.24. „Apply repeat time“ bude checkbox pre výber režimu merania, ktorý ak sa zaškrtnie, tak sa údaje zo spektrometra budú agregovať za čas definovaný užívateľom a ak sa odškrtnie, tak za počet, taktiež definovaný užívateľom.

3.1.25. Počet bodov meria (points) v shield buddy bude pole s natvrdo nastavenou kapacitou 8000 bodov. To znamená, že vieme maximálne namerať hodnotu v 8000 rôznych bodoch, než by sme ich poslali našej aplikácie. Je to ochrana kvôli tomu, aby sme nepresiahli kapacitu pamäte, keď ukladáme tieto hodnoty do poľa.

3.1.26. Cyklus merania je hodnota v mikrosekundách určujúca periódu merania spektrometra, hodnota je v rozsahu 0-20 000 mikrosekúnd. Cyklus merania sa vyráta vzorcom $\text{sampling} * \text{points}$, čo ale nesmie presiahnuť hranicu 20 000 mikrosekúnd.

3.2. Požiadavky rozhrania

Systém bude desktopová aplikácia. Bude použiteľná na počítači, ktorého displej má rozlíšenie aspoň 1366 × 768 bodov.

4. Návrh

4.1 Účel

Tento dokument slúži ako dôkladný technický návrh spôsobu implementácie informačného systému pre IMS spektrometer. Je určený primárne pre vývojový tím ako referenčný dokument, ale aj pre zadávateľa na kontrolu.

4.2 Rozsah

Informačný systém opísaný v tomto dokumente bude slúžiť na spracovanie údajov z IMS spektrometra. Jeho cieľom je zabezpečiť spracovanie nameraných údajov v podobe grafov. Jeho pridaná hodnota spočíva hlavne v zaznamenaní, sprehľadnení a zhromaždení údajov na jednom mieste.

4.3 Definície a skratky

Už uvedené v katalógu požiadaviek

5. Špecifikácia vonkajších interfejsov

5.1. Spektrometer

Spektrometer slúži na skúmanie prvkového chemického zloženia látky na základe merania vlnovej dĺžky odrazeného svetla od danej látky.

5.2. ShieldBuddy TC275

Shieldbuddy TC275 bude riešiť komunikáciu systému so spektrometrom. Bude slúžiť na nastavenie spektrometra podľa požadovaných parametrov zadanych do systému a spustenie merania. Taktiež bude následne čítať namerané hodnoty zo spektrometra a posilať ich do systému. Tieto prvky už boli funkčné v predchádzajúcom systéme, tak sme ich nemenili.

5.3. Meranie iónovej mobility

Model	ShieldBuddy TC275
Signál output	5V
Maximálny čas merania	20 milisekúnd
Hustota nameraných bodov (sampling)	1 - 40 (mikrosekundy)
Iónové zrážanie (gate)	1 – 20 (mikrosekundy)

6. Dátový model

6.1. Formáty súborov

Bez rozdielu na obsah údajov, všetky formáty ukladania budú v CSV (hodnoty oddelené čiarkou). Pre grafy, CSV súbor bude obsahovať riadky zodpovedajúce jednotlivým záznamom rozdelené pomocou oddeľovacieho znaku.

Budú obsahovať presne dva stĺpce pri agregovaných údajoch, kde prvý stĺpec je čas v mikrosekundách a druhý je zaznamenaný signál v bode. Pre intenzitný graf budeme mať počet stĺpcov zodpovedajúcich počtu agregovaných údajov v nich. Dáta sa budú zobrazovať v podobe grafu v užívateľskom rozhraní.

Formát údajov pre nastavenia a mobilitu bude dvojbodkou oddelený názov atribútu a jej hodnota.

Typy súborov budú:

- **Súbor s agregovanými údajmi** – obsahuje agregované údaje za čas definovaný užívateľom
- **Súbor s intenzitnými údajmi** – obsahuje viaceré agregované údaje
- **Súbor s nastaveniami merania** – obsahuje nastavenia aplikácie a mobilitu

6.2. Komunikačné protokoly

3.2.1. Sériový port RS232

Komunikácia s mikrokontrolérom prebieha cez sériové rozhranie usb, tak, že jednotlivé bity prenášaných dát sú vysielané postupne za sebou (v sérii) po jednom páre vodičov v každom smere.

Arduino obsahuje sériovú triedu pre posielanie údajov na COM port pre hosťovský počítač. Štandardná sériová trieda riadená s Arduino IDE je SERIALASC pre ShieldBuddy.

Alokácia sériových kanálov je nasledovná:

SerialASC Arduino FDTI USB-COM micro USB

Serial1 RX1/TX1 Arduino J403 pins 17/16

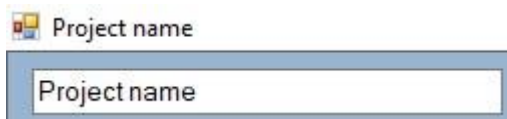
Serial0 RX0/TX0 Arduino J403 pins 15/14

Serial RX/TX Arduino default J402 pins D0/D1

7. Návrh používateľského rozhrania

Dizajn návrhu sa zmenil oproti ilustračnému obrázku v katalógu požiadaviek 3.1.1, tým, že po dohode s p. Matejčíkom (zadávatelom úlohy) sa sekcia „zrkadlový graf“ z úloh vynecháva.

7.1. Názov projektu



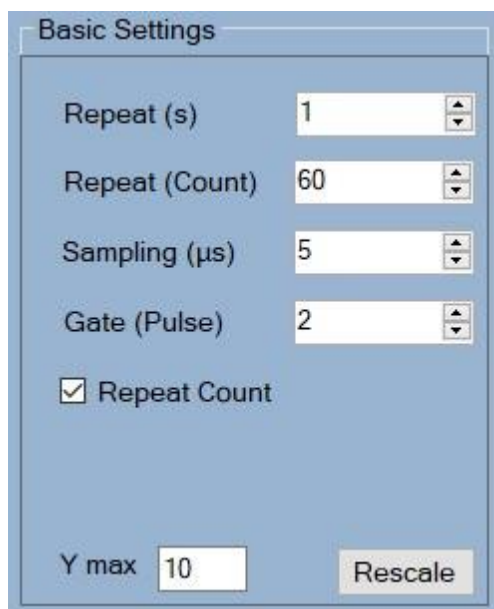
Meno projektu sa mení dynamicky, to, čo užívateľ napíše do textboxu bude meno projektu.

7.2. Nastavenia aplikácie

V danej sekcii užívateľ nastavuje parametre na základe ktorých bude ovplyvnený vypísaný graf, počet opakovaní v sekundách, počet meraní pre agregované dáta, impulz, ktorý spúšťa ióny do driftu v spektrometri a hustota merania.

Nastavujú sa parametre:

1. Repeat → zadávaný v sekundách (s)
2. Repeat → zadávaný počet opakovaní
3. Sampling → hustota merania, meraný v mý (μ s)
4. Gate → veľkosť impulzu
5. Repeat Count → checkbox prepínajúci repeat z bodu 1 alebo 2



Basic Settings

Repeat (s) 1

Repeat (Count) 60

Sampling (μ s) 5

Gate (Pulse) 2

☒ Repeat Count

Y max 10

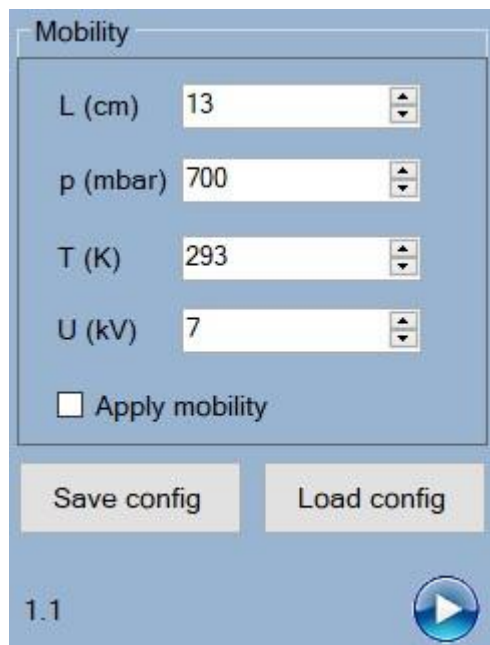
Rescale

Pri zvolení Repeat(Count) sa podľa zadaných parametrov vypočíta za aký čas prebehne simulácia. Táto informácia je len informačná pre užívateľa.

Grafu je možné aj zväčšiť či zmenšiť maximálnu hodnotu Y, hodnoty na osi X sú ovládané za pomoci myši.

7.3. Nastavenie Mobility

Užívateľ nastavuje hodnoty pre arduino, presnejšie za akých podmienok sa budú vykonávať merania spektrometra.



Mobility

L (cm) 13

p (mbar) 700

T (K) 293

U (kV) 7

☐ Apply mobility

Save config Load config

1.1

Play button

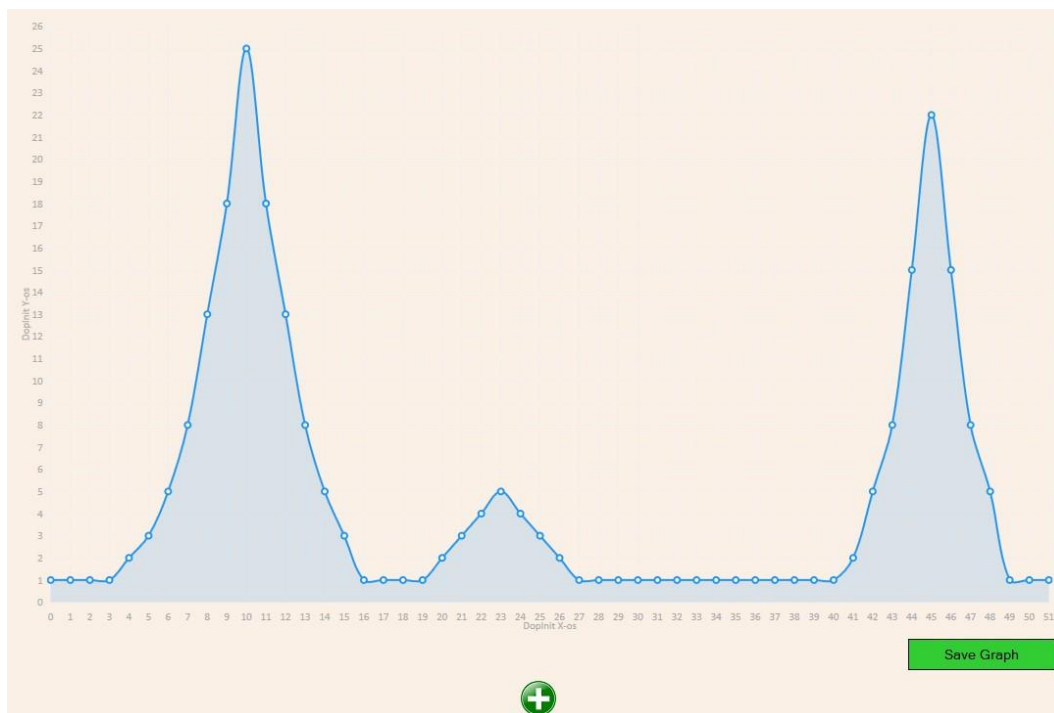
Nastavuje sa:

1. L \rightarrow dĺžka meraná centimetrom (cm)
2. P \rightarrow tlak meraný v mega baroch (mbar)
3. T \rightarrow teplota v kelvinoch (K)

4. $U \rightarrow$ elektrické napätie merané v kilo voltoch (kV)
5. Apply mobility \rightarrow aplikuje / neaplikuje mobilitu na dáta

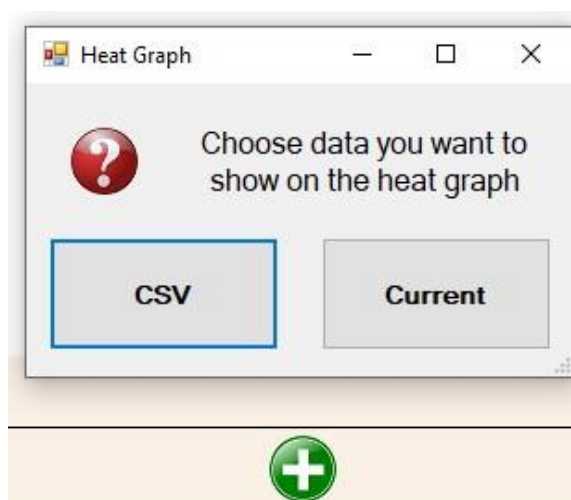
Okrem základných sekcií má užívateľ možnosť si uložiť konfiguráciu mobility pomocou tlačidla „Save config“, alternatívne možnosť načítať tieto údaje zo súboru tlačidlom „Load config“. Celé meranie sa spúšťa tlačidlom play, po ľavej strane od neho sa nachádza časovač, hodnoty sa budú rátať od 0 po nastavený Repeat zo sekcie Basic Settings.

7.4. Sekcia grafu

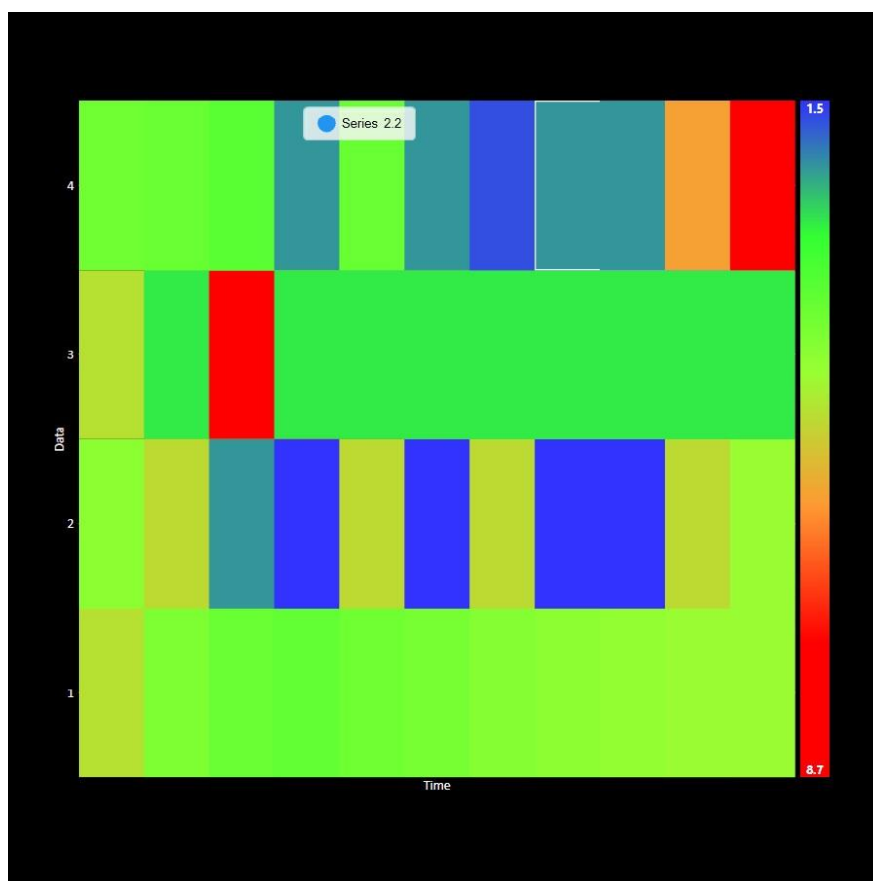


Na obrazovke je vykreslený graf z aktuálne nameraných dát. Jeho hodnoty a tvar sa menia v závislosti od parametrov spomenutých vyššie.

Tlačidlo plus ponúka možnosť vytvoriť intenzitný graf z aktuálne nameraných dát alebo už uložených do súboru *.csv.



Po úspešnom načítaní dát a grafu sa intenzitný graf pripne pod hlavný graf, každý jeden heat graf je možno zatvoriť tlačidlom „Cancel” a v prípade, že bol vytvorený z aktuálne nameraných dát pribudne aj tlačidlo Save na uloženie agregovaných dát do csv súboru a tlačidlo „Stop/Start” podľa toho, či je vykresľovanie zapnuté alebo vypnuté. Ak užívateľ klikne na ľubovoľný intenzitný graf vytvorí sa na pravej strane od jeho pozície graf obsahujúci údaje z daného riadku v zodpovedajúcom intenzitnom grafe.



8. Návrh implementácie

8.1. Class diagram

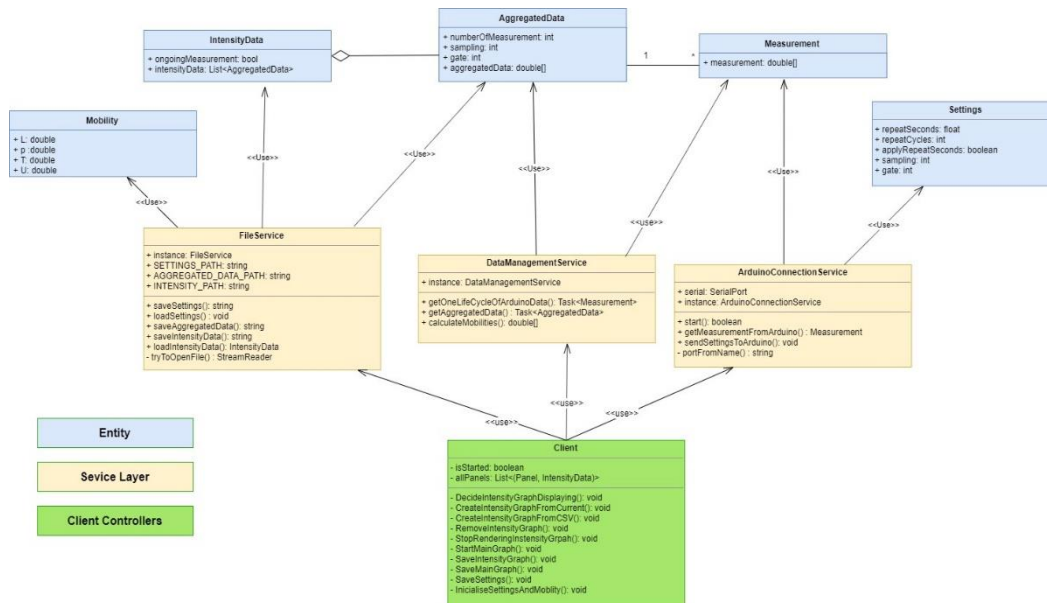
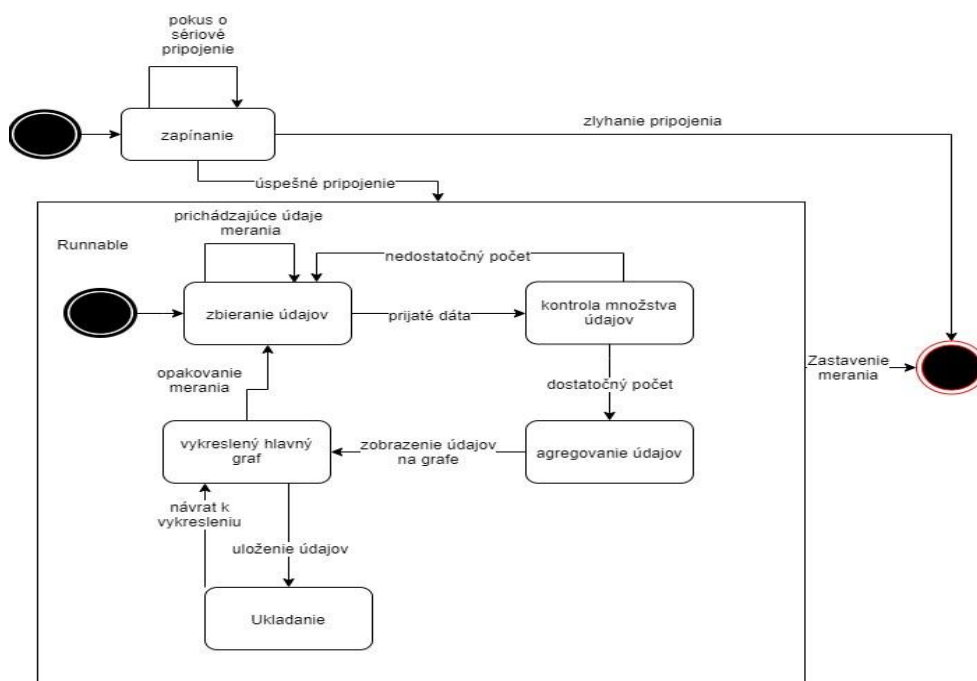


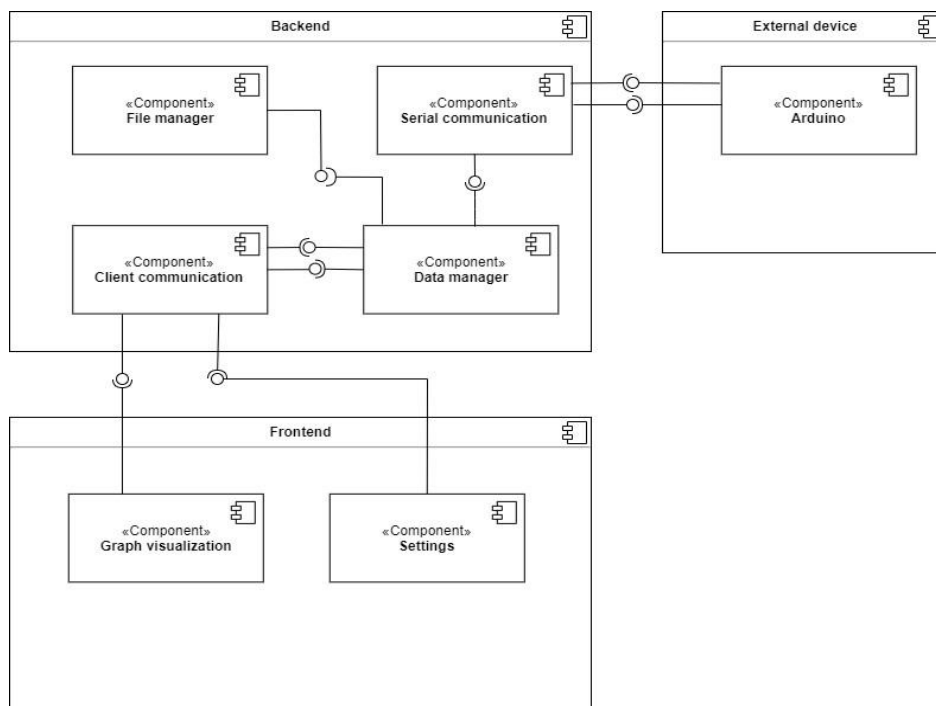
Diagram tried nasledujem architektúru zameranú na služby (service oriented architecture). Klient (zelená farba) je rozhranie komunikujúce s užívateľom, ktorý reaguje na diania (ako kliknutie na tlačidlo). Pre poskytnutie funkcionality nám pomáhajú služby (červená farba), ktoré fungujú ako singleton. Služby využívajú dátové štruktúry, entity (modrá farba) na uchovanie dočasných informácií.

8.2. Stavový diagram - Hlavný graf



Pri spustení aplikácie sa vytvorí pokus o sériové pripojenie k arduinu. Ak pripojenie zlyhá, aplikácie informuje chybovou hláškou užívateľa a končí. Ak sa pripojenie inicializuje, vytvorí sa vlákno, ktoré zbiera údaje z arduina. Hláškami Start a Stop arduino informuje začiatok a koniec merania. Pri správe stop sa skontroluje, či množstvo meraní presiahlo počet definovaným užívateľom. Ak áno, údaje sa agregujú a následne sa vykreslia v hlavnom grafe. Vlákno pokračuje v stave behu kým užívateľ nezastaví aplikáciu.

8.3. Diagram komponentov



Komponent diagram popisuje rozdelenie systému do samostatne fungujúcich častí. Základnou jednotkou nášho diagramu je backend a v ňom serial communication komponent, ktorý automaticky vyhľadá pripojené zariadenie (arduino), pokúsi s ním nadviazať spojenie a reaguje na príjem údajov. Prijaté údaje sa ďalej posielajú na data manager komponent, ktorý ich spracuje a využíva client communication komponent pre vykreslenie grafov v UI.

9. Testovacie scenáre

Dokument zahŕňa testovacie scenáre pre kontrolu správneho fungovania softwar-u.

Testy sú označené číslami od 1 po 13.

Kroky, ktoré sú nevyhnutné pre správne dokončenie testu sú označené písmenami z anglickej abecedy.

Ak niektorý bod v teste môže vyvolať neočakávané správanie aplikácie, napríklad vyhodíť výnimku, tieto vedľajšie účinky sú označené písmenami i. , ii. atď.

Ak v postupnosti bodov konkrétneho testu sa nevyvolá vedľajšie správanie aplikácie, test pokračuje ďalej.

Všetky výnimky, ktoré aplikácia vyhadzuje sa v GUI odchytiť, následne aplikácia zobrazí chybovú hlášku používateľovi ako pop-up a aplikácia pokračuje ďalej bez padnutia.

9.1. Test pripojenia prístroja a nastavenia portu

- a) Po spustení sa aplikácia pokúsi automaticky nájsť prístroj a jeho príslušný port.
- b) Port sa podarí detekovať a aplikácia dokáže správne interpretovať načítané hodnoty z prístroja.

i.

Ak sa port sa nepodarí detekovať, aplikácia vyhodí výnimku

ApplicationException,

9.2. Test nového merania

- a) Používateľ klikne na tlačidlo „Start“
- b) Aplikácia začne snímať prichádzajúce pole údaje na sériovom porte
- c) Aplikácia čaká na adekvátny počet meraní podľa počtu, alebo času definovaným užívateľom
- d) Po nazbieraní adekvátneho počtu meraní aplikácia vytvorí priemer údajov
- e) Spriemerované údaje sa nám zobrazia na hlavnom grafe
- f) Hlavný graf sa nám bude vykresľovať s novými spriemerovanými údajmi pokiaľ nebude ukončené stlačením tlačidla „Stop“

9.3. Test zmeny nastavení

- a) Pri zmene nastavení „gate“ a „sampling“ aplikácia odošle nové hodnoty prístroja
- b) Pri zmene nastavenia „repeat (s)“ sa zmení časový interval spriemerovanie meraní
- c) Pri zmene nastavenia „repeat (count)“ sa zmení početový interval sa spriemerovanie meraní
- d) Pri zaškrtnutí checkboxu „apply count“ sa aplikuje „repeat (count)“ na spriemerovanie údajov

9.4. Test uloženia nastavení

- a) Po stlačení tlačidla „save configuration“ sa nám uložia aktuálne nastavenie aplikácie a mobilita do csv súboru pod názvom, ktorý užívateľ zadal ako názov projektu a aktuálnym timestampom.
 - a. Ak aplikácii sa nepodarí uložiť údaje vyhodí výnimku Exception

- 9.5. Test načítania nastavení
- Používateľ zvolí tlačidlo „load configuration“
 - Zobrazí sa mu prieskumník súborov s možnosťou vybrať už uložený konfiguračný súbor
 - Používateľ zvolí žiadaný súbor
 - Konfiguračný súbor prepíše aktuálne nastavenia aplikácie a mobility načítanú zo súboru
 - Ak prepísanie údajov zlyhá aplikáciu vyhodí výnimku `FileLoadException`
- 9.6. Test uloženia hlavného grafu
- Používateľ zvolí tlačidlo „Save graph“ pod hlavným grafom
 - Aktuálne vykreslené údaje na hlavnom grafe sa uložia do csv súboru pod názvom, ktorý užívateľ zadal ako názov projektu a aktuálnym timestampom.
 - Ak sa aplikácii nepodarí uložiť údaje vyhodí výnimku `Exception`
- 9.7. Test možností generovania intenzitného grafu
- Po stlačení tlačidla „generuj intenzitný graf“ sa užívateľovi zobrazí pop-up s možnosťami „Načítanie zo súboru“ a „Generovanie z hlavného grafu“
- 9.8. Test generovania intenzitného grafu z hlavného grafu
- Používateľ zvolí možnosť „Generovanie z hlavného grafu“
 - Zobrazí sa mu intenzitný graf pod hlavným grafom s prázdnyimi údajmi
 - Všetky agregované údaje na hlavnom grafe sa zobrazia aj v intenzitnom grafe
 - Používateľ zastaví proces tlačidlom „Stop“ pod týmto intenzitným grafom
- 9.9. Test generovania intenzitného grafu zo súboru
- Používateľ zvolí možnosť „Načítanie zo súboru“
 - Zobrazí sa mu prieskumník súborov s možnosťou vybrať csv súbor s intenzitnými údajmi
 - Používateľ si zvolí žiadaný súbor
 - Ak aplikácia nedokáže otvoriť súbor vyhodí výnimku `FileNotFoundException`
 - Aplikácie sa pokúsi sparsovať údaje zo súboru do objektu `IntensityData`
 - Ak aplikácia nedokáže sparsovať súbor do objektu vyhodí výnimku `FileLoadException`
 - Aplikácia vykreslí intenzitný graf z načítaného súboru
- 9.10. Test uloženia intenzitného grafu
- Používateľ klikne na tlačidlo „Save“ pod intenzitným grafom generovaného z hlavného grafu
 - Aplikácia uloží tieto údaje do csv súboru pod názvom, ktorý užívateľ zadal ako názov projektu a aktuálnym timestampom.
 - Ak sa aplikácii nepodarí uložiť údaje vyhodí výnimku `Exception`
- 9.11. Test zmeny farieb intenzity
- Aplikácia po zobrazení intenzitného grafu zobrazí RGB škálu farieb pre daný graf.
 - Používateľ kliknutím na farbu môže zmeniť zafarbenie intenzity údajov v danom rozsahu, alebo zmeniť rozsah údajov spadajúcich do jednej farby

9.12. Test zobrazenia zrkadlového grafu

- a) Po zobrazení intenzitného grafu, používateľ klikne na niektorý bod v intenzitnom grafe
- b) Aplikácia zobrazí vedľa intenzitného grafu Agregovaný graf z ktorého daná úsečka v intenzitnom grafe vznikla

9.13. Test aplikovania mobility

- a) Používateľ vyplní hodnoty premenných pre aplikovanie mobility do textového podľa
- b) Používateľ zaškrtnie tlačidlo „Apply mobility“
- c) Aplikácia zoberie hodnoty z textového podľa a pretransformuje X-ovú os hlavného grafu, podľa bodu 3.1.10 v katalógu požiadaviek