

Návrh

Vision Lab – fyzikálne experimenty

Skupina SEJ2

Soňa Senkovičová, Erik Szalay, Jozef Kubík, Juraj Vetrák

23.11.2018

Obsah dokumentu

1. ÚVOD	3
1.1 ÚČEL DOKUMENTU	3
1.2 POUŽÍVANÉ DEFINÍCIE, AKRONYMY A SKRATKY	3
2. ŠPECIFIKÁCIA VONKAJŠÍCH INTERFEJSOV	3
2.1 KAMERA	3
3. FORMÁTY SÚBOROV	4
3.1 AKTUÁLNA SNÍMKA Z WEBOVEJ KAMERY V POZASTAVENOM ZÁZNAMU – JPG	4
3.2 AKTUÁLNY ZÁZNAM GRAFU - PNG	4
3.3 EXPORT DOKUMENTU Z POZASTAVENÉHO ZÁZNAMU – PDF	5
3.4 EXPORT SUROVÝCH DÁT Z MERANIA – CSV	6
3.5 IMPORT AJ EXPORT KONFIGURAČNÉHO SÚBORU – TXT	6
4. POUŽÍVATEĽSKÉ ROZHRAŇOVANIE	7
4.1 HLAVNÁ OBRAZOVKA	7
4.2 SPUSTENÉ MERANIE	7
4.3 EXPORTOVANIE ÚDAJOV	8
4.4 KOMENTÁR POUŽÍVATEĽA	8
4.5 NASTAVENIE GRAFU A KYVADLA	9
4.6 NASTAVENIE WEBOVEJ KAMERY	9
4.7 POPIS	10
5. NÁVRH IMPLEMENTÁCIE	10
5.1 PREHĽAD POUŽÍVANÝCH TECHNOLOGIÍ	10
5.1.1 Programovací jazyk	10
5.1.2 Knižnica pre manipuláciu s počítačovou grafikou	10
5.1.3 Knižnice pre tvorbu užívateľského prostredia	11
5.1.4 Knižnica pre generovanie PDF dokumentov	12
5.2 ARCHITEKTÚRA APLIKÁCIE	12
5.3 DIAGRAMY	13
5.3.1 Komponentový diagram	13
5.3.2 Triedny diagram	14
5.3.3 Stavový diagram	15
5.4 ROZDELENIE NA ČASTI (MODULY)	16
5.5 CIEĽOVÉ PROSTREDIE NASADENIA DO PREVÁDZKY	16

1. Úvod

1.1 Účel dokumentu

Účelom tohto dokumentu je špecifikovať formu a technologické detaily vývoja aplikácie Vision Lab - Fyzikálne experimenty. Dokument je určený pre vývojárov aplikácie.

1.2 Používané definície, akronymy a skratky

	Popis
PDF	Portable Document Format - súborový formát pre tvorbu dokumentov
JPG/JPEG	Metóda pre stratovú kompresiu digitálnych obrázkov
PNG	Portable Network Graphics - bezstratový formát rastrovej grafiky
CSV	Comma-separated Values - súborový formát vo forme čistého textu
TXT	Jednoduchý textový súbor
OpenCV	Open-source Computer Vision
highgui	High-level GUI and Media I/O

2. Špecifikácia vonkajších interfejsov

2.1 Kamera

Okrem výstupných súborov popísaných v časti [\[3\]](#) bude aplikácia komunikovať a prijímať vstupy z kamery. Môže ísť o internú kameru alebo kameru pripojenú cez USB. Komunikácia bude zabezpečená prostredníctvom systémového drivera a jeho ovládanie bude zabezpečené aplikačne prostredníctvom knižnice OpenCV a konfiguračného súboru [\[3.5\]](#).

3. Formáty súborov

3.1 Aktuálna snímka z webovej kamery v pozastavenom zázname – JPG

Načítanie obrázku (screenshotu) z kamery:

```
#include <opencv2/opencv.hpp>

using namespace cv;

int main( int argc, char** argv ) {
    char* imageName = argv[1];
    Mat image;
    image = imread( imageName, 1 );

    if ( argc != 2 || !image.data ) {
        printf( " No image data \n " );
        return -1;
    }

    imshow( imageName, image ); // prípadné zobrazenie obrázku

    ...
}
```

Uloženie obrázku (screenshotu) z kamery:

```
...

imwrite( "../images/imageName.jpg", image );
```

Uloženie obrázku sa realizuje do preddefinovaného priečinka, odkiaľ ho bude načítavať libHaru [\[3.3\]](#).

3.2 Aktuálny záznam grafu - PNG

Na tejto sekcii sa momentálne pracuje.

3.3 Export dokumentu z pozastaveného záznamu – PDF

Po inštalácii libHaru [\[5.1.4\]](#) do vývojového prostredia môže vývojár inicializovať objekt vytváraného dokumentu (HPDF_Doc), prípadne odchytiť chybové hlášky nasledovne:

```
#include "hpdf.h"

HPDF_Doc pdf;
pdf = HPDF_New (error_handler, NULL);
```

Vytvorenie novej stránky dokumentu:

```
HPDF_Page page_1;

page_1 = HPDF_AddPage (pdf);
```

Písanie textu (začiatok):

```
HPDF_STATUS HPDF_Page_BeginText (HPDF_Page page_1);
```

Písanie textu:

```
HPDF_STATUS HPDF_Page_TextOut (HPDF_Page page_1,
                                HPDF_REAL xpos, // pozícia x
                                HPDF_REAL ypos, // pozícia y
                                const char *text);
```

V libHaru sú začiatkové pozície pre x, y (0, 0) a nachádzajú sa v ľavom dolnom rohu dokumentu.

Písanie textu (ukončenie):

```
HPDF_STATUS HPDF_Page_EndText (HPDF_Page page);
```

Uloženie dokumentu na disk:

```
HPDF_SaveToFile (pdf, "test.pdf");
```

Pokračovanie v tvorbe nového dokumentu:

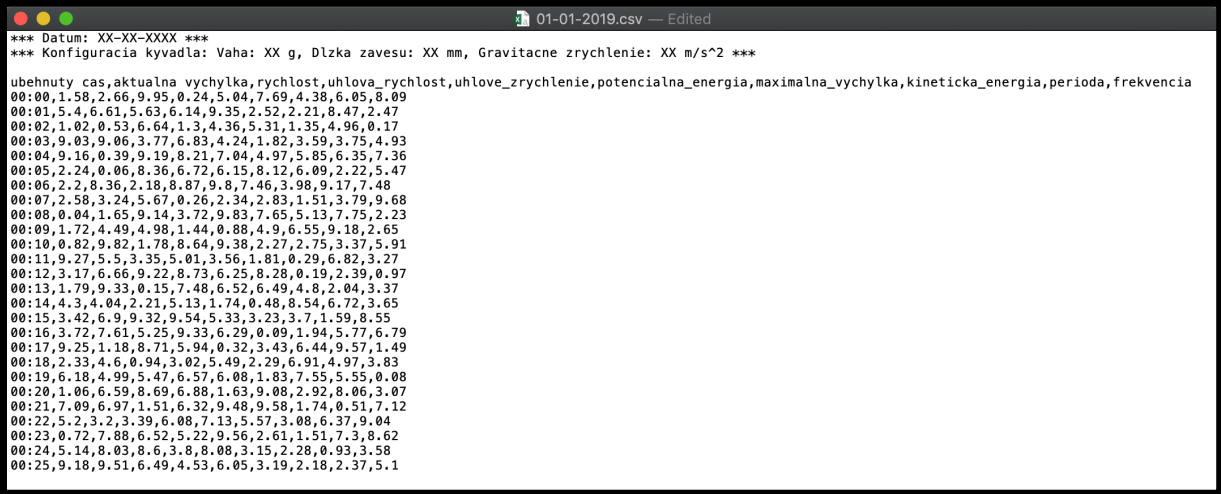
```
HPDF_NewDoc (pdf); // odstráni pôvodný
```

Uvoľnenie všetkých zdrojov:

```
HPDF_Free (pdf);
```

3.4 Export surových dát z merania – CSV

Formát vygenerovaného súboru počas spusteného merania:

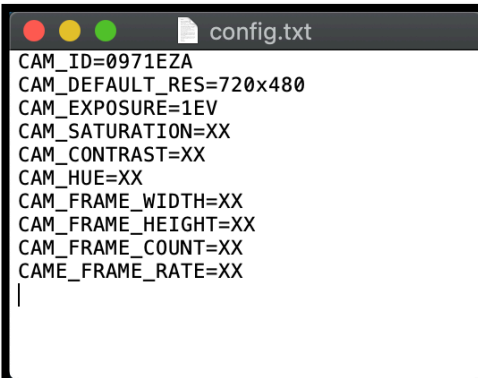


```
*** Datum: XX-XX-XXXX ***
*** Konfiguracia kyvadla: Vaha: XX g, Dlzka zavesu: XX mm, Gravitacne zrychlenie: XX m/s^2 ***
ubehnuty_cas,aktualna_vychylka,rychlost,uholova_rychlost,uholove_zrychlenie,potencialna_energia,maximalna_vychylka,kineticka_energia,perioda,frekvencia
00:00,1.58,2.66,9.95,0.24,5.04,7.69,4.38,6.05,8.09
00:01,5.4,6.61,5.63,6.14,9.35,2.52,2.21,8.47,2.47
00:02,1.02,0.53,6.64,1.3,4.36,5.31,1.35,4.96,0.17
00:03,9.03,9.06,3.77,6.83,4.24,1.82,3.59,3.75,4.93
00:04,9.16,0.39,9.19,8.21,7.04,4.97,5.85,6.35,7.36
00:05,2.24,0.06,8.36,6.72,6.15,8.12,6.09,2.22,5.47
00:06,2.2,8.36,2.18,8.87,9.8,7.46,3.98,9.17,7.48
00:07,2.58,3.24,5.67,0.26,2.34,2.83,1.51,3.79,9.68
00:08,0.04,1.65,9.14,3.72,9.83,7.65,5.13,7.75,2.23
00:09,1.72,4.49,4.98,1.44,0.88,4.9,6.55,9.18,2.65
00:10,0.82,9.82,1.78,8.64,9.38,2.27,2.75,3.37,5.91
00:11,9.27,5.5,3.35,5.01,3.56,1.81,0.29,6.82,3.27
00:12,3.17,6.66,9.22,8.73,6.25,8.28,0.19,2.39,0.97
00:13,1.79,9.33,0.15,7.48,6.52,6.49,4.8,2.04,3.37
00:14,4.3,4.04,2.21,5.13,1.74,0.48,8.54,6.72,3.65
00:15,3.42,6.9,9.32,9.54,5.33,3.23,3.7,1.59,8.55
00:16,3.72,7.61,5.25,9.33,6.29,0.09,1.94,5.77,6.79
00:17,9.25,1.18,8.71,5.94,0.32,3.43,6.44,9.57,1.49
00:18,2.33,4.6,0.94,3.02,5.49,2.29,6.91,4.97,3.83
00:19,6.18,4.99,5.47,6.57,6.08,1.83,7.55,5.55,0.08
00:20,1.06,6.59,8.69,6.88,1.63,9.08,2.92,8.06,3.07
00:21,7.09,6.97,1.51,6.32,9.48,9.58,1.74,0.51,7.12
00:22,5.2,3.2,3.39,6.08,7.13,5.57,3.08,6.37,9.04
00:23,0.72,7.88,6.52,5.22,9.56,2.61,1.51,7.3,8.62
00:24,5.14,8.03,8.6,3.8,8.08,3.15,2.28,0.93,3.58
00:25,9.18,9.51,6.49,4.53,6.05,3.19,2.18,2.37,5.1
```

Riadky súboru sú prebežne generované z triedy Dataset, z ktorej čerpá okrem tohto aj samotný GraphView.

3.5 Import aj export konfiguračného súboru – TXT

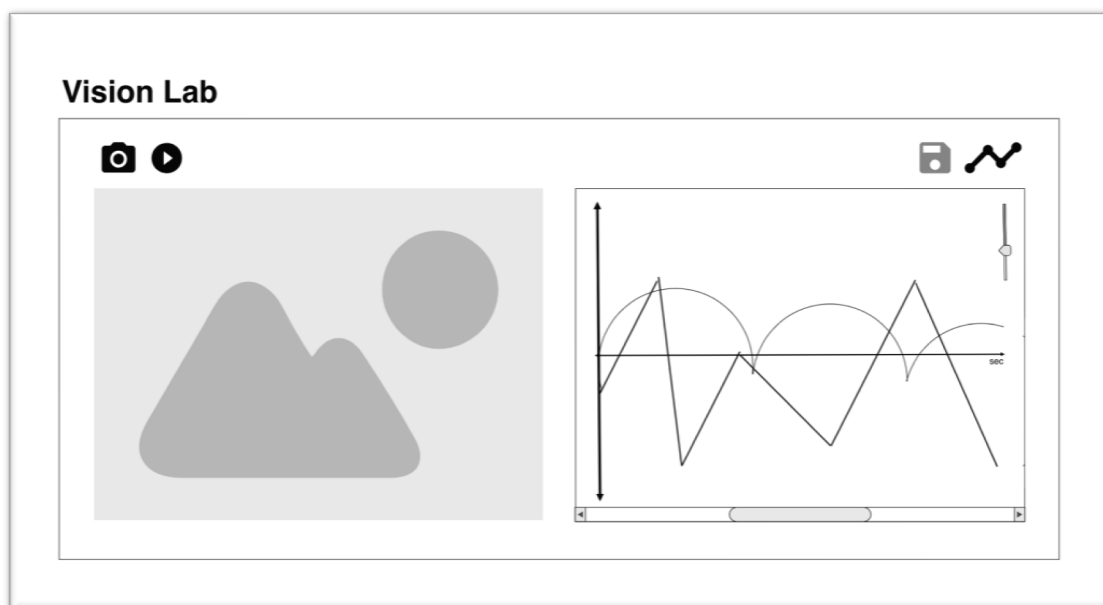
Formát konfiguračného súboru:



```
CAM_ID=0971EZA
CAM_DEFAULT_RES=720x480
CAM_EXPOSURE=1EV
CAM_SATURATION=XX
CAM_CONTRAST=XX
CAM_HUE=XX
CAM_FRAME_WIDTH=XX
CAM_FRAME_HEIGHT=XX
CAM_FRAME_COUNT=XX
CAM_FRAME_RATE=XX
|
```

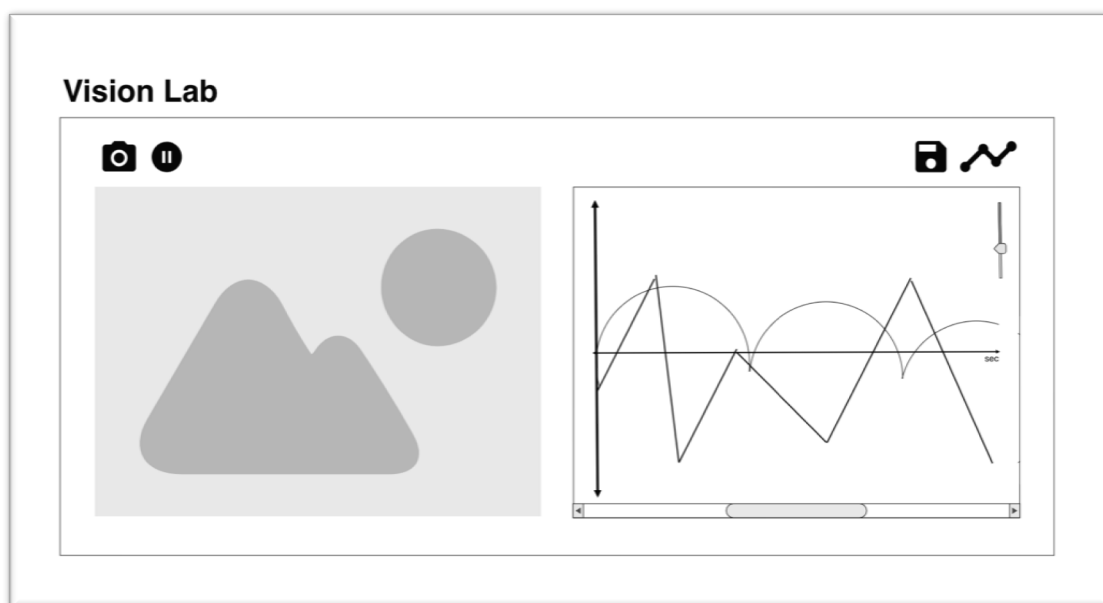
4. Používateľské rozhranie

4.1 Hlavná obrazovka



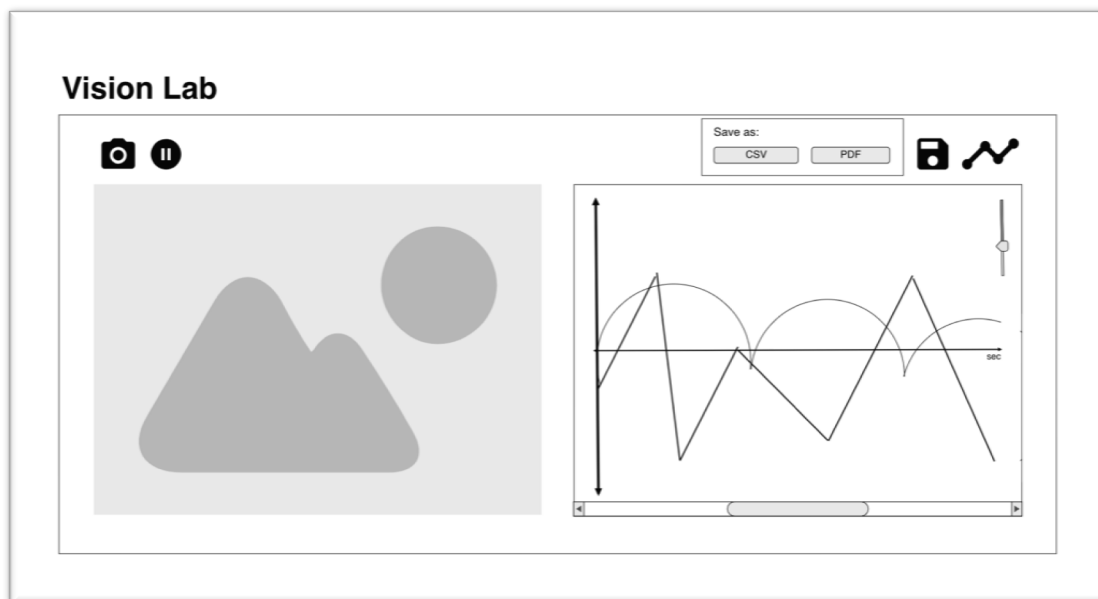
obr. 4.1 - hlavná obrazovka

4.2 Spustené meranie



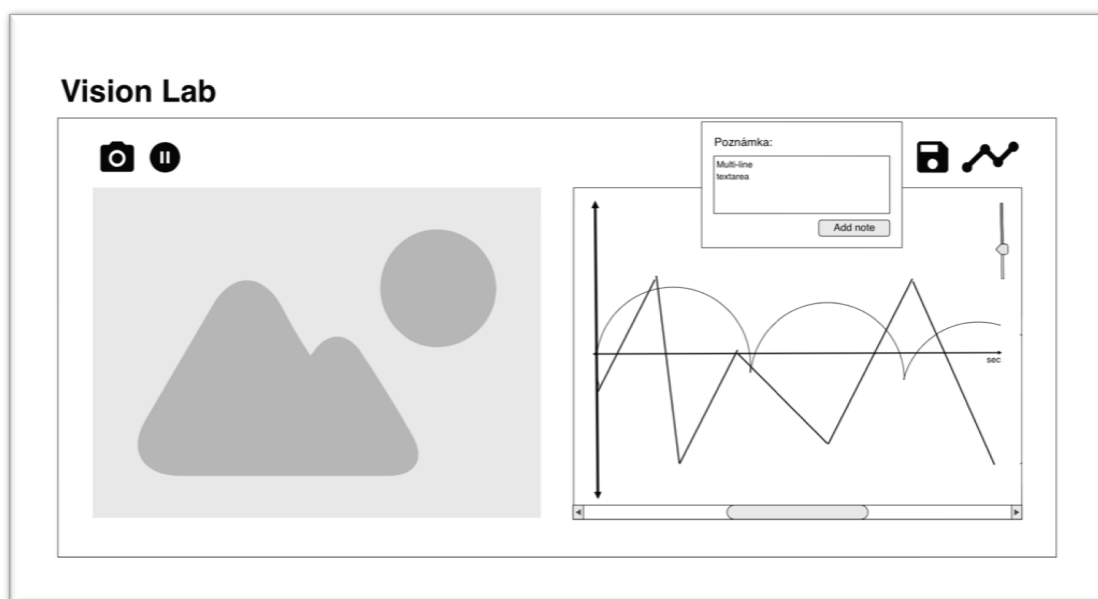
obr. 4.2 - spustené meranie

4.3 Exportovanie údajov



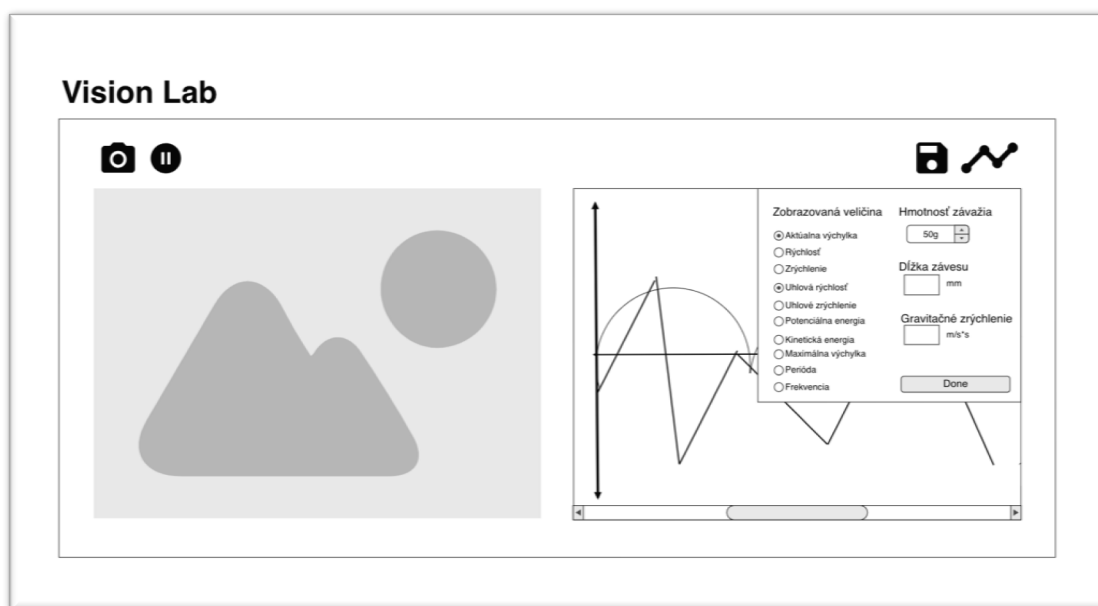
obr. 4.3 - exportovanie údajov

4.4 Komentár používateľa



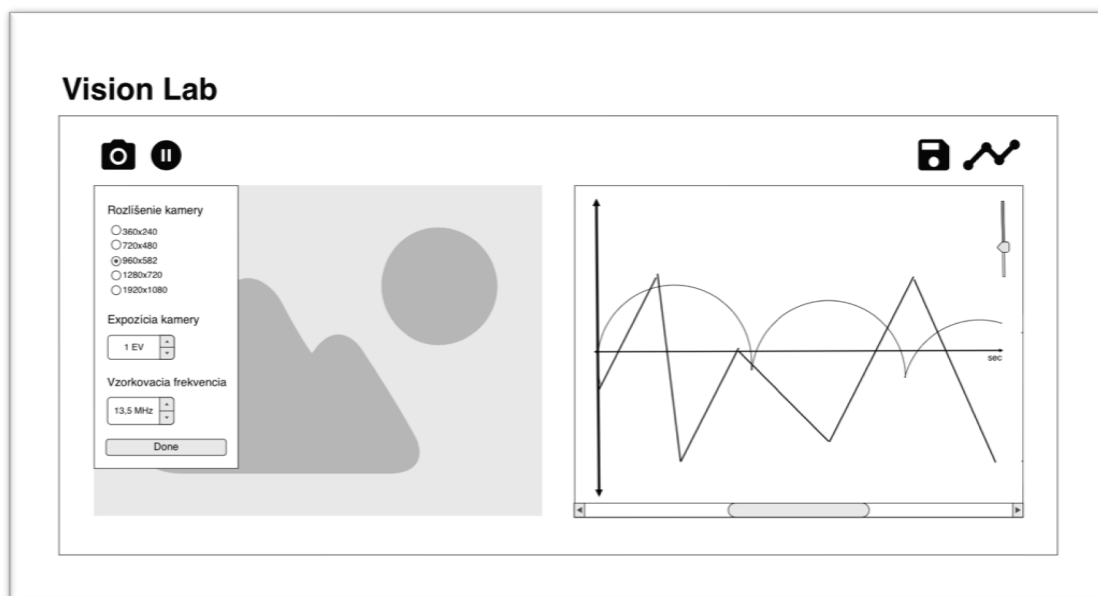
obr. 4.4 - pridanie komentáru používateľa

4.5 Nastavenie grafu a kyvadla



obr. 4.5 - nastavenie grafu a kyvadla

4.6 Nastavenie webovej kamery



obr. 4.6 - nastavenie webovej kamery

4.7 Popis

Aplikácia umožňuje sledovať záznam pohybu kyvadla a zároveň sledovať vybraný údaj (obr 4.1). Užívateľ má možnosť zastaviť a znovu spustiť záznam (obr. 4.2). Môže si vyexportovať údaje do formátov CSV (štatistické údaje) alebo PDF (rozsiahlejší dokument [3.3] (obr 4.3). Užívateľ si vie nastaviť, ktorý údaj chce pozorovať. Vedľa vie zadať hmotnosť závažia, dĺžku lanka a gravitačné zrýchlenie oblasti v ktorej je. Pomocou scrollbaru sa dá sledovať aj história grafu. Graf sa dá priblížiť alebo oddialiť (obr 4.5). Užívatelia majú možnosť nastaviť rozlíšenie a expozíciu kamery. Okrem toho vedia nastaviť aj frekvenciu snímania (obr 4.6).

5. Návrh implementácie

V tejto sekcii je podrobne popísaný návrh celkovej implementácie softvéru pozostávajúci z prehľadu používaných technológií, detailného popisu softvérovej architektúry, nakreslených diagramov popisujúcich funkčnosť aplikácie, rozdelenie na časti (moduly) a popis cieľového prostredia pri nasadení do prevádzky.

5.1 Prehľad používaných technológií

5.1.1 Programovací jazyk

Ako programovací jazyk bol zvolený C++, keďže predstavuje základný jazyk pre knižnicu OpenCV, ktorá bola použitá pre manipuláciu s kamerou a počítačovou grafikou.

C++ je objektovo orientovaný programovací jazyk so širokým spektrom dostupných knižníc. Keďže C++ predstavuje rozšírenú verziu C, ktorá je pomerne nízkoúrovňová, poskytuje oveľa väčšiu rýchlosť oproti vysokoúrovňovým jazykom ako Python alebo Java.

5.1.2 Knižnica pre manipuláciu s počítačovou grafikou

OpenCV (Open Source Computer Vision, <https://opencv.org/>) je knižnica pre manipuláciu s obrázkami a real-time videom. Je napísaná v C++, ktorý je aj jej primárnym interface jazykom.

OpenCV spracováva video ako sadu obrázkov (frameov). Tieto obrázky sú transformované do matice Mat, ktorá sa skladá z 2 častí:

- hlavičku (header) - obsahuje informácie o veľkosti (počet riadkov a stĺpcov), formát, v akom je obrázok uložený (napr. RGB alebo HSV), počet bitov pre každú hodnotu, či je signed, koľko hodnôt je na jeden pixel a pod.
- samotnú maticu (resp. pointer na ňu), ktorá predstavuje 2D pole s hodnotami, ktoré reprezentujú farbu každého bodu

Aby sa optimalizovala práca s veľkým objemom dát, Mat reálne obsahuje len header a pointer na samotné dáta, čím umožňuje zdieľanie rovnakých dát medzi metódami. Pointer môže ukazovať len na časť veľkého obrázka.

Manipulácia s videom v OpenCV je zabezpečená prostredníctvom triedy VideoCapture. Táto trieda umožňuje načítavanie videa zo súboru alebo z video-streamu a jeho transformáciu na jednotlivé frame-y, ako aj samotné ovládanie kamery a jej atribútov.

Modul highgui umožňuje základnú komunikáciu s GUI. Umožňuje zobrazovanie obrázkov aj framov z videa, buď v pôvodnej alebo upravenej verzii (napr. po zvýraznení určitého objektu alebo jeho stredu).

OpenCV obsahuje veľké množstvo metód na spracovanie obrázkov, vrátane štrukturálnej analýzy obrázkov a identifikácie objektov, ktorá dokáže identifikovať objekty rôznych tvarov (napr. funkcia moments vie identifikovať polygonálne objekty do 3. rádu). fitEllipse nájde elipsu obkolesujúcu 2D objekt.

5.1.3 Knižnice pre tvorbu užívateľského prostredia

Pre tvorbu užívateľského prostredia bola zvolená knižnica Microsoft Foundation Class (MFC) library. Ide o C++ knižnicu od Microsoftu (msdn.microsoft.com/en-us/library/d06h2x6e.aspx) pre vývoj desktopových aplikácií pre Windows.

Samotná MFC obaľuje časti tried Windows API v C++, vrátane funkcionalít, ktoré jej umožňujú využívať štandardný aplikačný framework. Obsahuje triedy pre ovládanie mnohých Windows objektov, predefinovaných okien a štandardných kontrolných prvkov.

5.1.4 Knižnica pre generovanie PDF dokumentov

libHaru (<http://libharu.org>) je open-source knižnica pre C++, určená na generovanie PDF dokumentov. Pre účely vyvíjaného softvéru sú dôležité nasledovné funkcie:

- Do generovaného PDF sa dajú umiestniť obrázky, riadky textu, poprípade odkazy.
- Podporuje vkladanie obrázkov vo formáte PNG, resp. JPG.
- Podporuje ukladanie vygenerovaného PDF dokumentu na špecifikované miesto na disku.

Pre Windows je dostupná vo forme dvoch typov vývojových prostredí:

- Static library - žiadne spustiteľné súbory nie sú potrebné
- Shared library (libhpdf.dll)

Inštalácia pre Windows:

```
unzip libharu-X.X.X.zip  
cd libharu-X.X.X
```

Microsoft VC++ Compiler:

```
nmake -f script/Makefile.msvc (or Makefile.bcc32_dll)
```

Otestovanie:

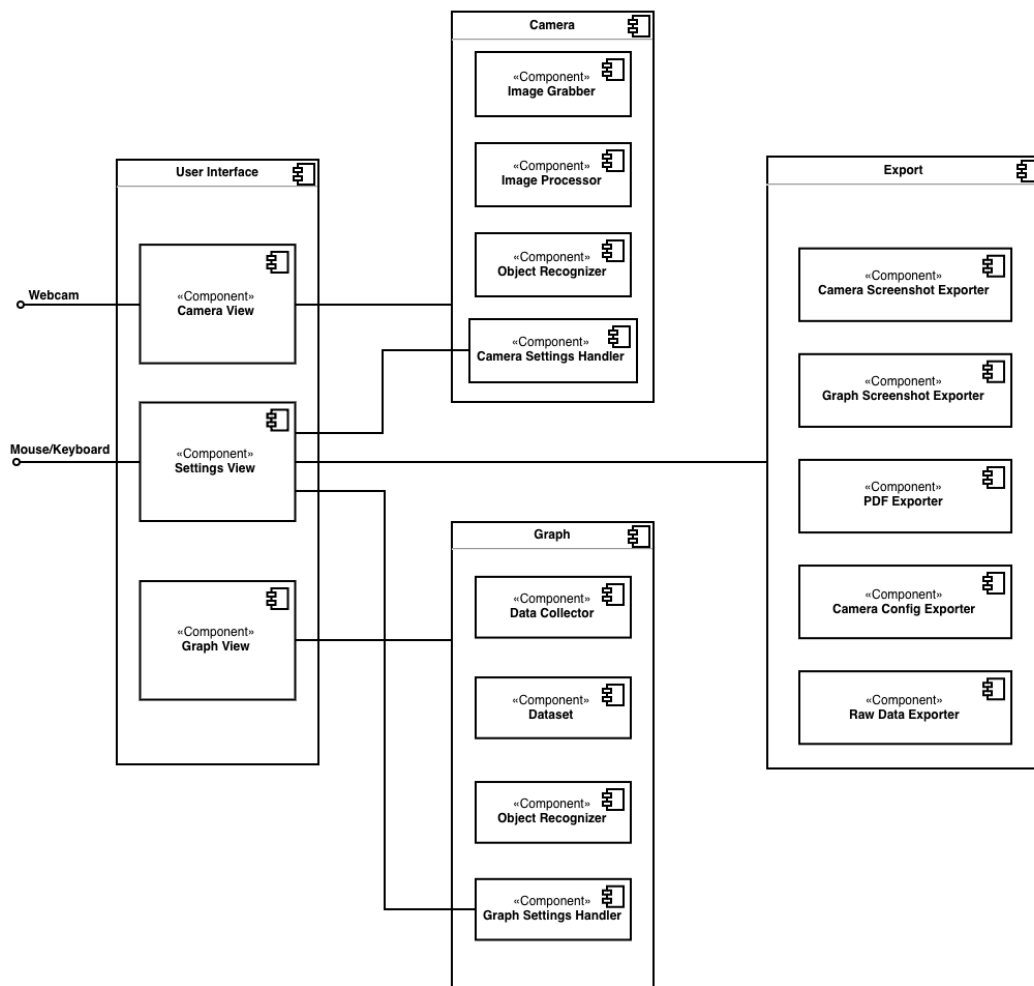
```
nmake -f script/Makefile.msvc[_dll] demo
```

5.2 Architektúra aplikácie

Na tejto sekcii sa momentálne pracuje.

5.3 Diagramy

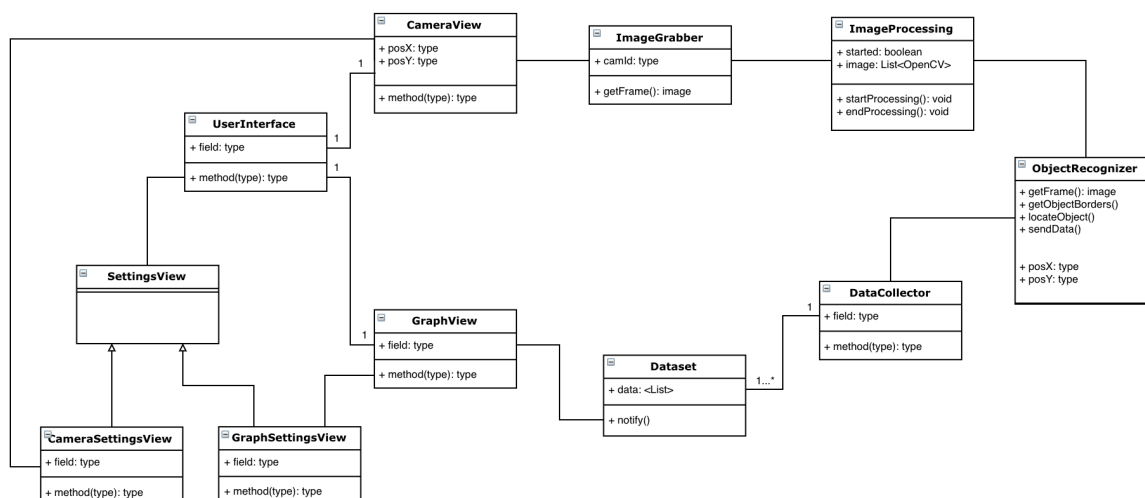
5.3.1 Komponentový diagram



obr 5.3.1 - Komponentový diagram

Obrázok 5.3.1 predstavuje diagram komponentov aplikácie. Komponent User Interface prepája všetky ostatné komponenty tak, že komunikujú prostredníctvom neho. Zobrazuje video z kamery prostredníctvom komponentu Camera. Ten spracováva kamerový záznam a snímané objekty (kyvadlo). Posiela obrázkové dáta komponentu Graph, ktorý vypočíta cez Data Collector príslušný Dataset. Ten komponent User Interface vykresľuje v podobe grafu. Cez komponent Settings dostávajú komponenty Camera a Graph informácie o užívateľských preferenciách a prostredníctvom Handlerov upravujú svoje nastavenia. Kompletný export údajov má na starosti komponent Export. Prípadné užívateľské komentáre dostáva z komponentu User Interface.

5.3.2 Triedny diagram



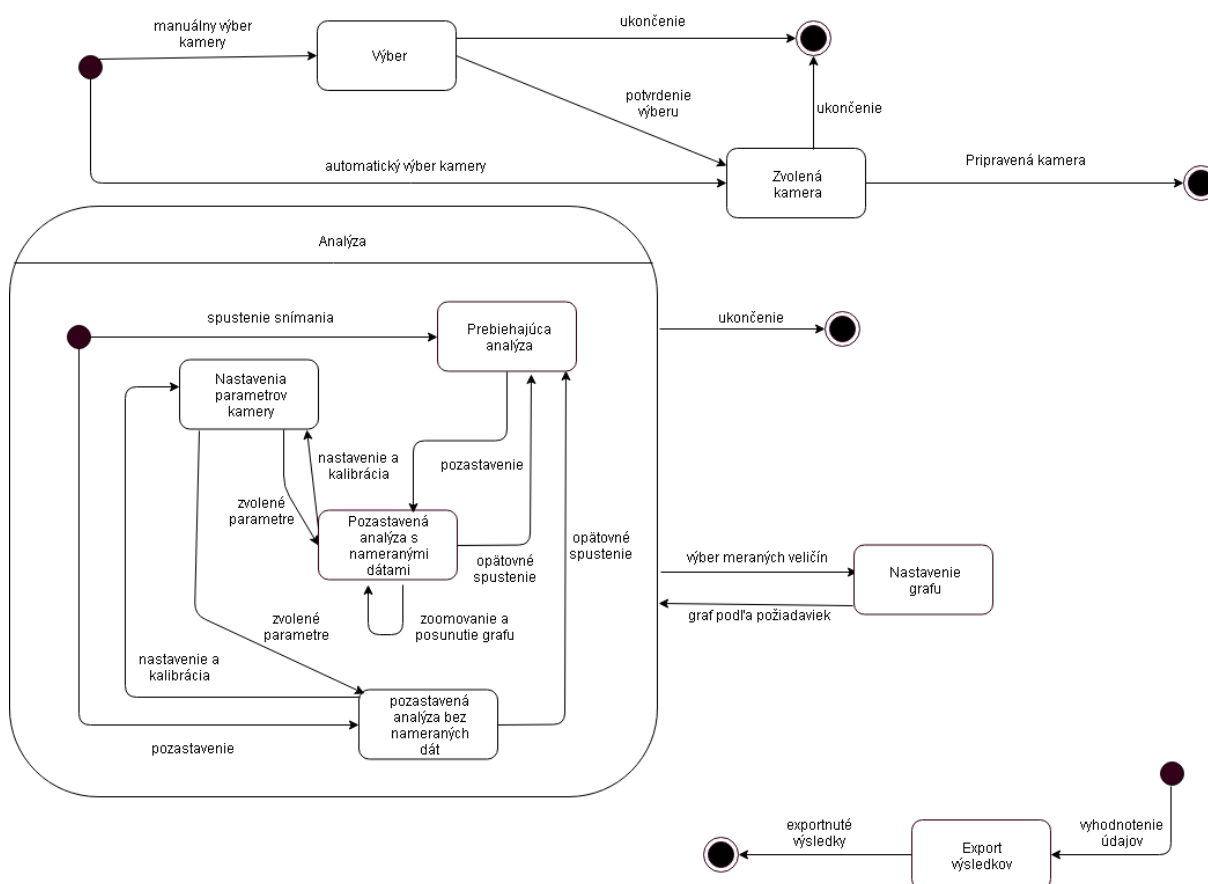
obr 5.3.2 - Triedny diagram

Obrázok 5.3.2 predstavuje triedny diagram vyvíjaného softvéru.

Základnou/hlavnou triedou je v tomto prípade `UIInterface`. Triedy `CameraView`, `GraphView` a `SettingsView` pracujú priamo s užívateľom, kde sa `CameraView` stará o zobrazovanie záznamu z webovej kamery, `GraphView` o vykresľovanie dát z `Dataset` do grafu a `SettingsView` bude spracovávať užívateľské vstupy a nastavovať rôzne komponenty.

ImageGrabber preberá jednotlivé snímky z kamery a ImageProcessing ich spracuje podľa nastavených parametrov kamery. ObjectRecognizer vyhľadá kyvadlo, DataCollector extrahuje fyzikálne dáta zo snímania kyvadla a vytvorí Dataset.

5.3.3 Stavový diagram



obr 5.3.3 - Stavový diagram

Obrázok 5.3.3 predstavuje 3 stavové diagramy, ktoré spolu popisujú rôzne stavy, v ktorých sa aplikácia môže nachádzať.

Prvý diagram zobrazuje stav hneď po spustení programu. Užívateľ má možnosť vybrať si medzi manuálnym a automatickým výberom kamery. Toto je dôležité z dôvodu, že môže byť pripojených viacero kamier a užívateľ musí byť v tom prípade na výber. Samozrejme, hocikedy sa môže užívateľ rozhodnúť program ukončiť.

Druhý diagram zobrazuje proces samotnej analýzy. Tu sa užívateľ dostane po úspešnom výbere kamery. Užívateľ môže hneď zapnúť snímánie, alebo ešte predtým sa rozhodnúť zmeniť nastavenia vybranej kamery, ako napríklad frekvenciu či rozlíšenie a až potom začať snímánie. Po začatí snímania môže užívateľ kedykoľvek túto činnosť pozastaviť, a opäť upraviť nastavenia kamery. Počas spustenej, ale aj pozastavenej analýzy sa taktiež vykresľuje v okne graf s nameranými hodnotami pomocou snímania. Tie hodnoty môže užívateľ kedykoľvek zmeniť podľa svojho uváženia. Samotný graf sa dá taktiež upravovať, konkrétne dá sa priblížiť alebo posunúť. Kedykoľvek počas týchto činností, zahrnutých v jednom makrostave, sa užívateľ môže samozrejme rozhodnúť ukončiť snímánie.

Tretí diagram zobrazuje stav po skončenej analýze. Údaje, ktoré boli pomocou snímania namerané, sa môže užívateľ rozhodnúť exportnúť pre neskoršie použitie alebo štúdium.

5.4 Rozdelenie na časti (moduly)

Na tejto sekcii sa momentálne pracuje.

5.5 Cieľové prostredie nasadenia do prevádzky

Systém bude využívaný na školách a seminároch, kde hlavnou témou bude sledovanie kyvadla a fyzikálnych javov týkajúcich sa jeho pohybu. Nevyhnutnou súčasťou prostredia je kyvadlo (ľahko viditeľné) na bielom pozadí a kamera naň namierená, zapojená do počítača. Systém umožňuje výber medzi dostupnými kamerami, takže pohyb kyvadla môže zaznamenávať aj viacero kamier.

Systém môže byť využívaný učiteľmi alebo prednášajúcimi - priamo na hodine s kyvadlom alebo len cez exportované údaje z aplikácie. Pri prednášaní na hodine vie učiteľ zobrazovať rôzne (aj viaceré) namerané veličiny z pohybu kyvadla. Údaje na grafe vie priblížiť/oddialiť a taktiež vie ukázať priebeh funkcie od začiatku nahrávania. Tieto údaje vie exportovať do PDF alebo CSV formátov (spolu s jej/jeho poznámkou) a ďalej používať pri výučbe.

V prípade používania systému študentmi, študenti si budú môcť vyskúšať sledovanie pohybu kyvadiel rôznej váhy a z rôznych materiálov. Po exporte dát ich budú vedieť porovnávať a využívať pri projektoch/experimentoch. Dá sa prezerať história grafu od začiatku nahrávania a teda sledovať väčšie zmeny priebehu funkcie v čase, ktoré by nebolo možné odsledovať iba na okne s aktuálnou časťou grafu.

Systém vie byť využívaný pri výskumoch/experimentoch/iných prostrediach aj inými užívateľmi, než len študentmi a učiteľmi, keďže exportované údaje budú zahŕňať aj raw data.