**Narzędzia Pracy Grupowej – sprawozdanie z projektu**

**”Who’s There?”**

**Wymagania:**

1. Udostępnienie zasobów projektu:
   1. Produkt – aplikacja, kod, dokumentacja
   2. Narzędzia – GIThub (public), … – linki (sprawdzone): raporty, tablice, Wiki
2. Jedno wspólne sprawozdanie całego zespołu (z indywidualnie opisanym wkładem 3)
   1. Krótki opis organizacji pracy – metodyka, wymienione narzędzia, artefakty - ½ str.
   2. Dla kolejnych sprintów (newsletter) – role (SM), cel, kamień milowy, zadania, przydział, czasy realizacji – 2-3 str., zebrania
   3. Szczegółowy BACKLOG produktu (PBL) - ewolucja, priorytety, porządkowanie – 2 str.
3. Dla każdego wykonawcy zespołu (każdy przygotowuje dla siebie) – wkład – 2-3 str.:
   1. Imię, nazwisko, mail – dewelopera
   2. Zadania zrealizowane (projektowe, programistyczne, organizacyjne, dokumentacyjne) - wymienić
   3. Raporty - linki (standup, commit) – aktywność udokumentowana narzędziami (kalendarze, tabele, statystyki, wykresy)
   4. Parametry oceny pracy, …, procentowa ocena wkładu własnego w projekt (tabela)

# Założenia projektu:

Stworzenie aplikacji mającej na celu rozpoznawanie twarzy i otwieranie zamka do drzwi osobom zapisanym w programie.

Projekt będzie złożony z trzech komponentów:

Program robiący i zapisujący zdjęcia do folderu, program rozpoznający twarz osoby ze zdjęcia oraz program otwierający zamek na podstawie informacji z drugiego programu.

02.05.2024

Pierwsze spotkanie organizacyjne. Ustalono podstawowe założenia projektu: cel, język programowania (python) i główne importowane biblioteki (tkinter i openCV). Dodatkowo utworzono repozytorium i tablice projektu GitHub.

Obraz zawierający szkic, rysowanie, pismo odręczne, diagram

Opis wygenerowany automatycznieWstępny szkic działania programu ustalony na pierwszym spotkaniu

07.05.2024

Drugie spotkanie organizacyjne. Przetestowano sprawność repozytorium poprzez dodanie, edycje i usunięcie dowolnego pliku przez każdego członka zespołu. Dodatkowo ustalono pierwszy sprint w terminie od 7.05.2024 do 13.05.2024. Jako pierwszego Scrum Mastera wybrano Kamila Jędrzejko. W tablicy kanban projektu ustawiono zadania oraz przypisano je do wszystkich uczestniczących w projekcie. Ustalono, że związane problemy w trakcie wykonywania poszczególnych zadań będą umieszczane w ich komentarzach wraz z datą oraz ilością czasu poświęconego na wykonaną pracę.

Metodyka tworzenia projektu podczas jego trwania ulegała zmianie, dlatego została ona opisana w każdym ze sprintów.

# Pierwszy sprint(Iteration):

Jako cel pierwszego(tygodniowego) sprintu postawiono doprowadzenie aplikacji do podstawowej funkcji rozpoznawania twarzy tzn. stworzenie 3 głównych plików projektu w wybranym indywidualnie przez członków edytorze kodu. Zadania zostały przydzielone po wcześniejszym ustaleniu umiejętności członków zespołu w trakcie spotkania. Były one odpowiednio opisane, a w razie ewentualnych problemów członkowie kontaktowali się dowolnie z obecnym Scrum Masterem w celu ustalenia aktualnej wizji projektu.

Na tym etapie łatwość użytkowania oraz graficznego interfejsu użytkownika (GUI) nie są priorytetem. Dlatego podczas tego sprintu skupiliśmy się na funkcjonalności podstawowej. Z tego powodu, wszystkie prace były wykonywane bez tworzenia dodatkowych gałęzi w repozytorium GitHub, co pozwoliło na stworzenie podstawy programu jak i samego repozytorium. Naszym kamieniem milowym było ukończenie 3 osobnych programów współgrających ze sobą w gałęzi, które były umieszczane w gałęzi głównej programu ‘main’.

Zadania pierwszego sprintu wraz z opisem podano poniżej:

* **Robienie zdjęć: zaprogramowanie kamery**
* **Robienie zdjęć: Stworzenie warunku sprawdzającego if**
* **trening: enkodowanie**
* **main: Odczytanie danych z pliku .yaml**
* **main: szacowanie rozpoznania**
* **main: złączenie całości**

Poniżej przedstawiono również ilość zadań sprintu(7), progres oraz ilość godzin poświęconej na zadania.

Obraz zawierający zrzut ekranu, Oprogramowanie multimedialne, wyświetlacz, Oprogramowanie graficzne

Opis wygenerowany automatycznie

Rysunek 1 Ilość zadań każdego z członków zespołu

Obraz zawierający zrzut ekranu, oprogramowanie, Oprogramowanie multimedialne, wyświetlacz

Opis wygenerowany automatycznie

Rysunek 2 Progres pierwszego sprintu

Jak widać po statystykach dostępnych na repozytorium, nasz pierwszy sprint przebiegł pomyślnie, dzięki czemu mogliśmy zająć się kolejną częścią projektu.

Rysunek 3 Czas pracy Pierwszego sprintu

# Drugi sprint(Iteration2):

Celem drugiego sprintu zaplanowanego na dni 14-23 maja było dodanie podstawowych elementów interfejsu graficznego do aplikacji, tak aby przedstawić funkcjonalność programu bez konieczności korzystania z konsoli. Również w tej części projektu przyjęto metodykę scrumową przydzielając zadania każdemu z członków zespołu. Zadania polegały na dodaniu elementu interfejsu lub jego funkcjonalności. Gdyż edycja interfejsu zabiera możliwość testowania prototypu zastosowano osobne branche do kontynuacji prac nad każdym podprogramem. W gałęzi main kontynuowano dokumentację oraz pozostawiono wersję alfa programu.

Na tym etapie głównym celem było zamienienie opleceń wcześniej obsługiwanych przez konsole na najlepiej odpowiadające im elementy interfejsu. Ergonomia oraz styl nie były priorytetami tego sprintu. Wszystkie edycje kolorów i czcionek mają jedynie na celu zwiększenie czytelności elementów do komunikacji z użytkownikiem

Dodatkowo w tym sprincie dodano kilka zadań dotyczących organizacji projektu

Zadania drugiego sprintu wraz z opisem podano poniżej:

* main: dodać funkcjonalność przycisku do wychodzenia z programu
* trening: dodanie graficznego interfejsu
* trening: przypisanie funkcjonalności do interfejsu
* take\_photo: dodanie przycisków
* take\_photo: implementacja przycisków wyjścia i zapisu
* take\_photo: implementacja przycisku usuwania
* main: dodać wyraźne graficzne potwierdzenie uzyskania dostępu
* take\_photo: Dodanie graficznego licznika zdjęć
* trening: dodanie graficznej informacji rezultacie operacji
* main: dodać przycisk do opuszczania programu
* take\_photo: umieszczenie podglądu z kamery w oknie tkinter
* main: umieścić okno podglądu kamery w oknie tkinter
* take\_photo: dodanie graficznej informacji o pomyślnym zapisie zdjęcia
* Skompletowanie backlogu sprintu 1
* Dodanie backlogu
* take\_photo: dodanie pola tekstowego do wprowadzania nazwy użytkownika

Poniżej przedstawiono wykresy dotyczące wyników pracy w sprincie 2:

Obraz zawierający zrzut ekranu, Oprogramowanie graficzne, Oprogramowanie multimedialne, oprogramowanie

Opis wygenerowany automatycznie

Rysunek 3 Ilość zadań każdego z członków zespołu

Obraz zawierający zrzut ekranu, Oprogramowanie graficzne, Oprogramowanie multimedialne

Opis wygenerowany automatycznie

Rysunek 4 Progress drugiego sprintu

Rysunek 5 Czas pracy Drugiego sprintu

Założenia sprintu drugiego nie zostały osiągnięte. Każde nieukończone zadanie zostanie przeniesione do następnego sprintu. Stan aktualnych plików nie pozwala na merge do następnej iteracji prototypu.

# Trzeci sprint(Iteration3):

# Podsumowanie prac

Prace nad programem zakończone zostają na punkcie skompletowania podprogramów z wszystkich branchy do branchy main tworząc wersje prototypową programu nr. 2. Program jest wypełni funkcjonalny i spełnia założenia postawione na ten etap rozwoju.

Poniższy wykres przedstawia wkład pracy każdego z uczestników projektu

Rysunek 6 Czas pracy poświęcony na cały projekt

Obraz zawierający zrzut ekranu, tekst, Wykres, diagram

Opis wygenerowany automatycznie

Rysunek 7 Rozkład zadań w całym projekcie na dzień 09.06.2024

# Nasz zespół:

Deweloper: Kamil Jędrzejko / K4r0x / deox846@gmail.com

**Zadania projektowe:**

- wstępna struktura początkowa projektu zaliczeniowego

**Zadania programistyczne:**

- Zapoznać się z biblioteką python face\_recognition

- Zapoznać się z biblioteką python CV2

- Zapoznać się z biblioteką python tkinter

- main: złączenie całości

- take\_photo: Dodanie graficznego licznika zdjęć

- trening: dodanie graficznej informacji rezultacie operacji

- main: dodać przycisk do opuszczania programu

- Zmiana koloru przycisku usuwania zdjęć na czerwony

**Zadania dokumentacyjne:**

- utworzenie opisu użytkownika(plik README.md)

- stworzenie początkowego szkieletu dokumentacji projektu

- Skompletowanie backlogu sprintu 1

**Zadania organizacyjne:**

- inicjator 2 spotkania projektowego

|  |  |  |
| --- | --- | --- |
| Aspekt | Parametry | Wkład |
| Role | Wymienić | Scrum master pierwszego sprintu, organizator 2 spotkania, programista |
| Kodowanie | Liczba linii kodu | 437 ++ | 465 -- |
|  | Funkcje (wymienić) | Moja część kodu nie zawierała funkcji |
| Repozytorium | Liczba commit-ów | 7 |
|  | Liczba utworzonych gałęzi | 1 |
|  | Gałąź (używana – nazwa) | main, GUI\_main, GUI\_traning, GUI\_Take\_Photo |
|  | Liczba połączonych gałęzi | 1 |
|  | Liczba dni aktywności GIT | 6 |
| Dokumentowanie | Liczba standup-ów |  |
|  | Opisy na Wiki |  |
| Aktywność | Liczba zrealizowanych zadań | 10 |
|  | Szacowana liczba godzin | 7 |
|  | Ocena procentowego wkładu | 25% |

Poniżej przedstawiam moją aktywność w repozytorium, która potwierdza wykonanie przeze mnie przydzielonych mi zadań.

Obraz zawierający tekst, zrzut ekranu, oprogramowanie, Czcionka

Opis wygenerowany automatycznie

Rysunek X Wykonane zadania

Obraz zawierający tekst, zrzut ekranu, Oprogramowanie multimedialne, oprogramowanie

Opis wygenerowany automatycznie

Rysunek X Aktywność z liczbą commitów

Deweloper: Patryk Konieczny / konpferd / koniecznypatryk05@gmail.com

**Zadania projektowe:**

- wstępna wprowadzenie GUI oraz podstawowe zabezpieczenia projektu zaliczeniowego

**Zadania programistyczne:**

- Zapoznać się z biblioteką python face\_recognition

- Zapoznać się z biblioteką python CV2

- Zapoznać się z biblioteką python tkinter

- Robienie zdjęć: zaprogramowanie kamery

- take\_photo: implementacja przycisku usuwania

- take\_photo: umieszczenie podglądu z kamery w oknie tkinter

- main: umieścić okno podglądu kamery w oknie tkinter

- take\_photo: dodanie graficznej informacji o pomyślnym zapisie zdjęcia

- take\_photo: dodanie pola tekstowego do wprowadzania nazwy użytkownika

- skompletowanie prototypu programu w wersji 2

**Zadania dokumentacyjne:**

- utworzenie streszczenia pierwszych 2 spotkań organizacyjnych

- podsumowanie prac nad projektem w dokumentacji

**Zadania organizacyjne:**

- inicjator 1 spotkania projektowego

|  |  |  |
| --- | --- | --- |
| Aspekt | Parametry | Wkład |
| Role | Wymienić | Scrum master drugiego sprintu, organizator 1 spotkania, programista |
| Kodowanie | Liczba linii kodu | 612 ++ | 313 -- |
|  | Funkcje (wymienić) | callback(), save\_photo(), delete\_photos(), show\_widget, get\_existing\_names() |
| Repozytorium | Liczba commit-ów | 33 |
|  | Liczba utworzonych gałęzi | 3 |
|  | Gałąź (używana – nazwa) | main, GUI\_main, GUI\_traning, GUI\_Take\_Photo |
|  | Liczba połączonych gałęzi | 4 |
|  | Liczba dni aktywności GIT | 15 |
| Dokumentowanie | Liczba standup-ów | 14 |
|  | Opisy na Wiki |  |
| Aktywność | Liczba zrealizowanych zadań | 11 |
|  | Szacowana liczba godzin | 15,5 |
|  | Ocena procentowego wkładu | 40% |

Poniżej przedstawiam moją aktywność w repozytorium, która potwierdza wykonanie przeze mnie przydzielonych mi zadań.

Obraz zawierający tekst, zrzut ekranu, oprogramowanie, numer

Opis wygenerowany automatycznie

Rysunek 8 Wykonane zadania

Obraz zawierający tekst, zrzut ekranu

Opis wygenerowany automatycznie

Rysunek 9 Aktywność na github

Deweloper: Justyna Biegun / jbiegun14 / biegun.jb@gmail.com

**Zadania projektowe:**

- Interfejs graficzny oraz błędy programu

**Zadania programistyczne:**

- Zapoznać się z biblioteką python face\_recognition

- Zapoznać się z biblioteką python CV2

- Zapoznać się z biblioteką python tkinter

- Robienie zdjęć: Stworzenie warunku sprawdzającego if

- main: szacowanie rozpoznania

- main: dodać wyraźne graficzne potwierdzenie uzyskania dostępu

- Naprawienie problemu z błędnym wykrywaniem twarzy uciętej przez krawędź pola widzenia kamery

- trening: przypisanie funkcjonalności do interfejsu

**Zadania dokumentacyjne:**

- uzupełnienie dokumentacji z pierwszego spotkania organizacyjnego

- uzupełnienie backlogu

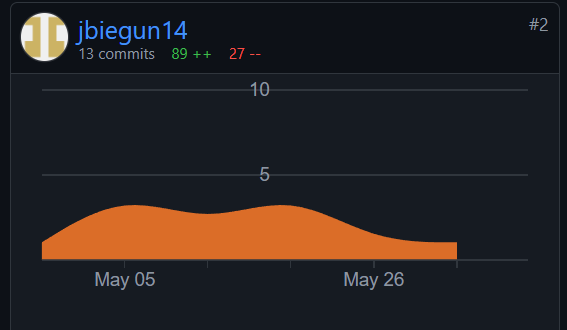
- Skompletowanie uwag z laboratoriów

**Zadania organizacyjne:**

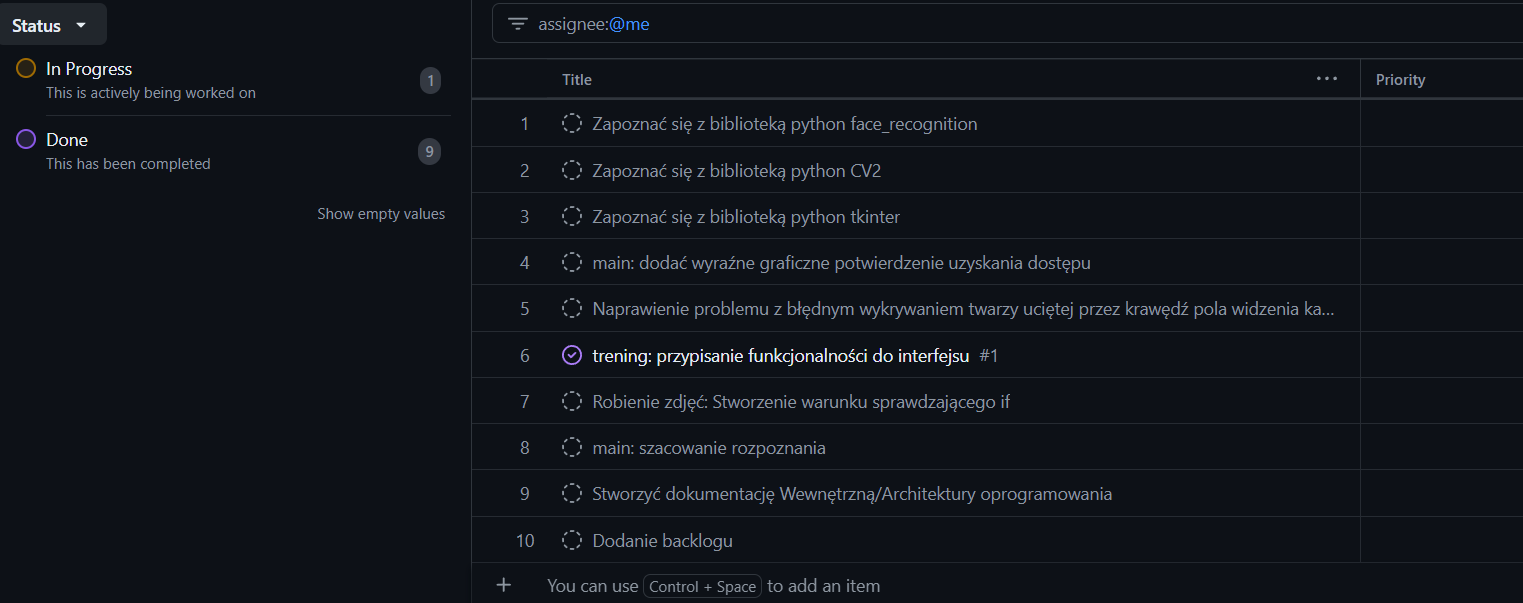
- brak

|  |  |  |
| --- | --- | --- |
| Aspekt | Parametry | Wkład |
| Role | Wymienić | Programista, częściowe uzupełnianie dokumentacji, częściowy wkład w organizację spotkań |
| Kodowanie | Liczba linii kodu | 89 ++ | 27 -- |
|  | Funkcje (wymienić) | gainAccess |
| Repozytorium | Liczba commit-ów | 13 |
|  | Liczba utworzonych gałęzi | 0 |
|  | Gałąź (używana – nazwa) | main, GUI\_main, GUI\_traning, GUI\_Take\_Photo |
|  | Liczba połączonych gałęzi | 0 |
|  | Liczba dni aktywności GIT | 10 |
| Dokumentowanie | Liczba standup-ów |  |
|  | Opisy na Wiki |  |
| Aktywność | Liczba zrealizowanych zadań | 10 |
|  | Szacowana liczba godzin | 7 |
|  | Ocena procentowego wkładu | 20% |

Aktywność w repozytorium:



Rysunek 8 Aktywność



Rysunek 9 Zadania

|  |  |  |
| --- | --- | --- |
| Aspekt | Parametry | Wkład |
| Role | Wymienić |  |
| Kodowanie | Liczba linii kodu |  |
|  | Funkcje (wymienić) |  |
| Repozytorium | Liczba commit-ów |  |
|  | Liczba utworzonych gałęzi |  |
|  | Gałąź (używana – nazwa) |  |
|  | Liczba połączonych gałęzi |  |
|  | Liczba dni aktywności GIT |  |
| Dokumentowanie | Liczba standup-ów |  |
|  | Opisy na Wiki |  |
| Aktywność | Liczba zrealizowanych zadań |  |
|  | Szacowana liczba godzin |  |
|  | Ocena procentowego wkładu |  |