**Змiст**

[**1.** **ЗАГАЛЬНІ ВІДОМОСТІ ПРО ОРГАНІЗАЦІЮ** 1](#_Toc148994413)

[**2. Інструктаж з техніки безпеки і охорони праці.** 2](#_Toc148994414)

[2.1 Загальні положення 2](#_Toc148994415)

[2.2 Вимоги до виробничих приміщень для експлуатації ПК 4](#_Toc148994416)

[2.3 Гігієнічні вимоги до параметрів виробничого середовища приміщень з ПК 5](#_Toc148994417)

[2.3.1 Мікроклімат 5](#_Toc148994418)

[2.3.2 Освітлення 5](#_Toc148994419)

[2.3.3 Шум і вібрація 6](#_Toc148994420)

[2.3.4 Неіонізуючі електромагнітні випромінювання 6](#_Toc148994421)

[2.4 Гігієнічні вимоги до організації і обладнання робочих місць з ПК 7](#_Toc148994422)

[2.5. Вимоги до режимів праці і відпочинку при роботі з ПК 8](#_Toc148994423)

[2.6 Вимоги до профілактичних медичних оглядів 9](#_Toc148994424)

[**3.Перелiк робiт.** 10](#_Toc148994425)

[3.1 Інсталяція та налагодження програмного забезпечення 10](#_Toc148994426)

[3.2 Робота з мовою програмування Python, бiблiотекою Flask, базою даних Postgres та Flutter Framework. 13](#_Toc148994427)

[3.2.2 Робота з Flutter Framework. 17](#_Toc148994428)

[3.2.3 Робота з Git та GitHub 23](#_Toc148994429)

[**ВИСНОВОК** 28](#_Toc148994430)

[**ПЕРЕЛІК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ** 29](#_Toc148994431)

# **ЗАГАЛЬНІ ВІДОМОСТІ ПРО ОРГАНІЗАЦІЮ**

Громадська організація "Харківський Кластер Інформаційних Технологій" є неприбутковою спілкою, що об'єднує провідні українські компанії в галузі інформаційних технологій, навчальні заклади і владні структури для спільного розвитку економіки, IT-сфери та подолання спільних викликів. На даний момент більше 200 організацій становлять часть цього кластера, включаючи такі компанії, як SoftServe, NIX solutions, Sigma Software, CodeIT, Promodo, AltexSoft і багато інших.

У партнерстві з освітніми установами, "Харківський IT Кластер" реалізує ряд освітніх проєктів з метою модернізації освітніх програм і залучення експертів з галузі IT.

Кластер діє в шести основних напрямках:

1. Підвищення рівня освіти населення для забезпечення якісними кадрами на ринку.
2. Розвиток міської інфраструктури.
3. Підвищення іміджу Харкова і його просування на міжнародному рівні.
4. Створення додаткової цінності для компаній-учасників Кластера.
5. Надання юридичної підтримки та захисту.
6. Реалізація благодійних ініціатив, в рамках проєкту IT4Life.

Адреса розташування: Громадянська вулиця, 11/13, Харків, Харківська область, 61000.

# **2. Інструктаж з техніки безпеки і охорони праці.**

## 2.1 Загальні положення

Правила поширюються на умови й організацію праці при роботі з персональним комп’ютером.

Ці правила призначені для запобігання несприятливої дії на працівників шкідливих факторів, які супроводжують роботу з ПК, пов'язаною з зоровим і нервово-емоційним напруженням, що виконується у вимушеній робочій позі при локальному напруженні верхніх кінцівок на фоні обмеженої загальної м'язової активності (гіподинамії) під впливом комплексу фізичних факторів шуму, електростатичного поля, неіонізуючих та іонізуючих електромагнітних випромінювань.

Відповідальність за виконання цих Правил покладається на посадових осіб, фізичних осіб, які займаються підприємницької діяльністю і здійснюють розробку, виробництво, закупівлю, реалізацію й застосування ПК в адміністративних і промислових приміщеннях.

Порушення санітарно-гігієнічних і санітарно-протиепідемічних правил і норм тягне дисциплінарну, адміністративну, кримінальну відповідальність відповідно до Закону України "Про забезпечення санітарного та епідеміологічного благополуччя населення" (ст.45, 46, 49).

## 2.2 Вимоги до виробничих приміщень для експлуатації ПК

Площа на одне робоче місце має становити не менше ніж 6,0 м2, а об'єм не менше ніж 20,0 м3.

Приміщення для роботи з ПК повинні мати природне та штучне освітлення відповідно до СНиП II-4-79.

Приміщення для роботи з ПК мають бути обладнані системами опалення, кондиціонування повітря, або припливно-витяжною вентиляцією відповідно до СНиП 2.04.05-91. Нормовані параметри мікроклімату, іонного складу повітря, вмісту шкідливих речовин мають відповідати вимогам СН 4088-86, СН 2152-80, ГОСТ 12.1.005-88, ГОСТ 12.1.007-76 та (дод.2,3).

Віконні прорізи приміщень для роботи з ПК мають бути обладнані регульованими пристроями (жалюзі, завіски, зовнішні козирки).

Забороняється для оздоблення інтер'єру приміщень ПК застосовувати полімерні матеріали (деревинно-стружкові плити, шпалери, що миються, рулонні синтетичні матеріали, шаруватий паперовий пластик тощо), що виділяють у повітря шкідливі хімічні речовини.

Виробничі приміщення можуть обладнуватись шафами для зберігання документів, магнітних дисків, полицями, стелажами, тумбами тощо з урахуванням вимог до площі приміщень.

У приміщеннях з ПК слід щоденно робити вологе прибирання.

Приміщення з ПК мають бути оснащені аптечками першої медичної допомоги.

При приміщеннях з ПК мають бути обладнані побутові приміщення для відпочинку під час роботи, кімната психологічного розвантаження.

## 2.3 Гігієнічні вимоги до параметрів виробничого середовища приміщень з ПК

### 2.3.1 Мікроклімат

У виробничих приміщеннях на робочих місцях з ПК мають забезпечуватись оптимальні значення параметрів мікроклімату: температури, відносної вологості й рухливості повітря (ГОСТ 12.1.005-88, СН 4088-86).

Рівні позитивних і негативних іонів у повітрі приміщень з ПК мають відповідати санітарно-гігієнічним нормам N 2152-80.

### 2.3.2 Освітлення

Штучне освітлення в приміщеннях з робочими місцями, обладнаними ПК, має здійснюватись системою загального рівномірного освітлення.

Зазначення освітлення освітленості на поверхні робочого столу в зоні розміщення документів має становити 300-500 лк. Якщо ці значення освітленості неможливо забезпечити системою загального освітлення, допускається використовувати місцеве освітлення. При цьому світильники

місцевого освітлення слід встановлювати таким чином, щоб не створювати бліків на поверхні екрана, а освітленість екрана має не перевищувати 300 лк.

Як джерела світла в разі штучного освітлення мають застосовуватись переважно люмінісцентні лампи типу ЛБ. У разі влаштування відбитого освітлення у виробничих та адміністративно-громадських приміщеннях допускається застосування металогалогенних ламп потужністю 250 Вт. Допускається застосування ламп розжарювання у світильниках місцевого освітлення.

Система загального освітлення має становити суцільні або преривчасті лінії світильників, розташовані збоку від робочих місць (переважно ліворуч), паралельно лінії зору працюючих.

Слід передбачити обмеження прямої блискості від джерел природного та штучного освітлення. При цьому яскравість світлих поверхонь (вікна, джерела штучного освітлення ), що розташовані в полі зору повинна бути не більше ніж 200 кд/м2.

Для забезпечення нормованих значень освітленості у приміщеннях з ПК слід чистити шибки і світильники принаймні двічі на рік і вчасно замінювати лампи, що перегоріли.

### 2.3.3 Шум і вібрація

Рівні звукового тиску в октавних смугах частот, рівні звуку та еквівалентні рівні звуку на робочих місцях, обладнаних ПК, мають відповідати вимогам СН 3223-85, ГОСТ 12.1.003-83, ГР 2411-81.

Устаткування, що становить джерело шуму (АЦП, принтери тощо), слід розташовувати поза приміщенням для роботи ПК.

Для забезпечення допустимих рівнів шуму на робочих місцях слід застосовувати засоби звукопоглинання, вибір яких має обгрунтовуватись спеціальними інженерно-акустичними розрахунками.

### 2.3.4 Неіонізуючі електромагнітні випромінювання

Значення напруженості електростатичного поля на робочих місцях з ПК (як у зоні екрана дисплея, так і на поверхнях обладнання, клавіатури,

друкувального пристрою) мають не перевищувати гранично допустимих за ГОСТ 12.1.045-84, СН 1757-77.

Значення напруженості електромагнітних полів на робочих місцях з ПК мають відповідати нормативним значенням (ГДР N 3206-85, ГДР N 4131-86, СН N 5802-91, ГОСТ 12.1.006-84).

Інтенсивність потоків інфрачервоного випромінювання має не перевищувати допустимих значень відповідно до СН 4088-86, ГОСТ 12.1.005- 88.

Іонізуючі електромагнітні випромінювання на відстані 0,05 м від екрана до корпуса відеотермінала при будь-яких положеннях регулювальних пристроїв не повинна перевищувати 7,74 x 10 в ст.-12 А/кг, що відповідає еквівалентній дозі 0,1 мбер/год (100 мкР/год) НРБУ N 58.

## 2.4 Гігієнічні вимоги до організації і обладнання робочих місць з ПК

Робочі місця з ПК слід так розташовувати відносно світових прорізів, щоб природнє світло падало збоку переважно зліва.

Конструкція робочого столу має відповідати сучасним вимогам ергономіки і забезпечувати оптимальне розміщення на робочій поверхні використовуваного обладнання (дисплея, клавіатури, принтера) і документів.

Висота робочої поверхні робочого столу з ПК має регулюватися в межах 680...800 мм, а ширина і глибина - забезпечувати можливість виконання операцій у зоні досяжності моторного поля (рекомендовані розміри: 600...1400 мм, глибина - 800..1000 мм).

Робочий стілець має бути підйомно-поворотним, регульованим за висотою, з кутом і нахилу сидіння та спинки і за відстанню від спинки до переднього краю сидіння поверхня сидіння має бути плоскою, передній край - заокругленим. Регулювання за кожним із параметрів має здійснюватися незалежно, легко і надійно фіксуватися.

Для зниження статичного напруження м'язів верхніх кінцівок слід використовувати стаціонарні або змінні підлокітники завдовжки не менше ніж 250 мм, завширшки 50...70 мм, що регулюються за висотою над сидінням у межах 230...260 мм і відстанню між підлокітниками в межах 350...500 мм.

Екран ПК має розташовуватися на оптимальній відстані від очей користувача, що становить 600...700 мм, але не ближче ніж за 600 мм з урахуванням розміру літерно-цифрових знаків і символів.

Клавіатуру слід розташовувати на поверхні столу на відстані 100...300 мм від краю, звернутого до працюючого.

Розташування пристрою введення - виведення інформації має забезпечувати добру видимість екрана ПК, зручність ручного керування в зоні досяжності моторного поля і за висотою - 900...1300 мм, за шириною 400...500 мм.

## 2.5. Вимоги до режимів праці і відпочинку при роботі з ПК

Внутрішньозмінні режими праці і відпочинку мають передбачати додаткові нетривалі перерви в періоди, що передують появі об'єктивних і суб'єктивних ознак втомлення і зниження працездатності.

При виконанні протягом дня робіт, що належать до різних видів трудової діяльності, за основну роботу з ПК слід вважати таку, що займає не менше 50% часу впродовж робочої зміни мають передбачатися:

- перерви для відпочинку і вживання їжі (обідні перерви);

- перерви для відпочинку і особистих потреб (згідно з трудовими нормами);

- додаткові перерви, що вводяться для окремих професій з урахуванням

особливостей трудової діяльності.

У всіх випадках, коли виробничі обставини не дозволяють застосувати регламентовані перерви, тривалість безперервної роботи з ПК не повинна перевищувати 4 години.

В окремих випадках - при хронічних скаргах працюючих з ПК на зорове втомлення незважаючи на дотримання санітарно-гігієнічних вимог до режимів праці і відпочинку, а також застосування засобів локального захисту очей -допускаються індивідуальних підхід до обмеження часу робіт з ПК, зміни характера праці, чергування з іншими видами діяльності, не пов'язаними з ПК.

Активний відпочинок має полягати у виконанні комплексу гімнастичних вправ, спрямованих на зняття нервового напруження, м'язове розслаблення, відновлення функцій фізіологічних систем, що порушуються протягом трудового процесу, зняття втоми очей, поліпшення мозкового кровообігу і працездатності.

За умови високого рівня напруженості робіт з ПК показане психологічне розвантаження у спеціально обладнаних приміщеннях (в кімнатах психологічного розвантаження) під час регламентованих перерв або в кінці робочого дня.

## 2.6 Вимоги до профілактичних медичних оглядів

Працюючі з ПК підлягають обов'язковим медичним оглядам: попереднім - при влаштуванні на роботу і періодичним - протягом трудової діяльності відповідно до наказу МЗ України N 45 від 31.03.94 р.

Періодичні методичні огляди мають проводитися раз на два роки комісією в складі терапевта, невропатолога та офтальмолога.

До складу комісії, що проводить попередні та періодичні медичні огляди, при необхідності можуть залучатись до оглядів лікарів інших спеціальностей.

Основними критеріями оцінки придатності до роботи з ПК мають бути показники стану органів зору: гострота зору, показники рефракції, акомодації, стану бінокулярного апарату ока тощо. При цьому необхідно враховувати також стан організму в цілому.

# **3.Перелiк робiт.**

## 3.1 Інсталяція та налагодження програмного забезпечення

Інсталяція (встановлення) - це процес встановлення програмного забезпечення на комп'ютері користувача під Mac OS. Більшість програм мають власну програму для інсталяції. Щоб розпочати інсталяцію, вам потрібно відкрити файл інсталяції в папці програми. Зазвичай цей файл має розширення .dmg або .pkg.

Під час інсталяції вам може бути запропоновано прийняти ліцензійну угоду та відповісти на питання щодо налаштувань. Ви можете вибрати тип інсталяції, який вам потрібний, такий як стандартний або змінений (призначений для користувачів, які бажають вибрати компоненти для встановлення).

Для прикладу, я встановив такі програми: Python, Postgres, pgAdmin 4 та Visual Studio Code. Також я встановив такі бібліотеки: psycopg2, flask.

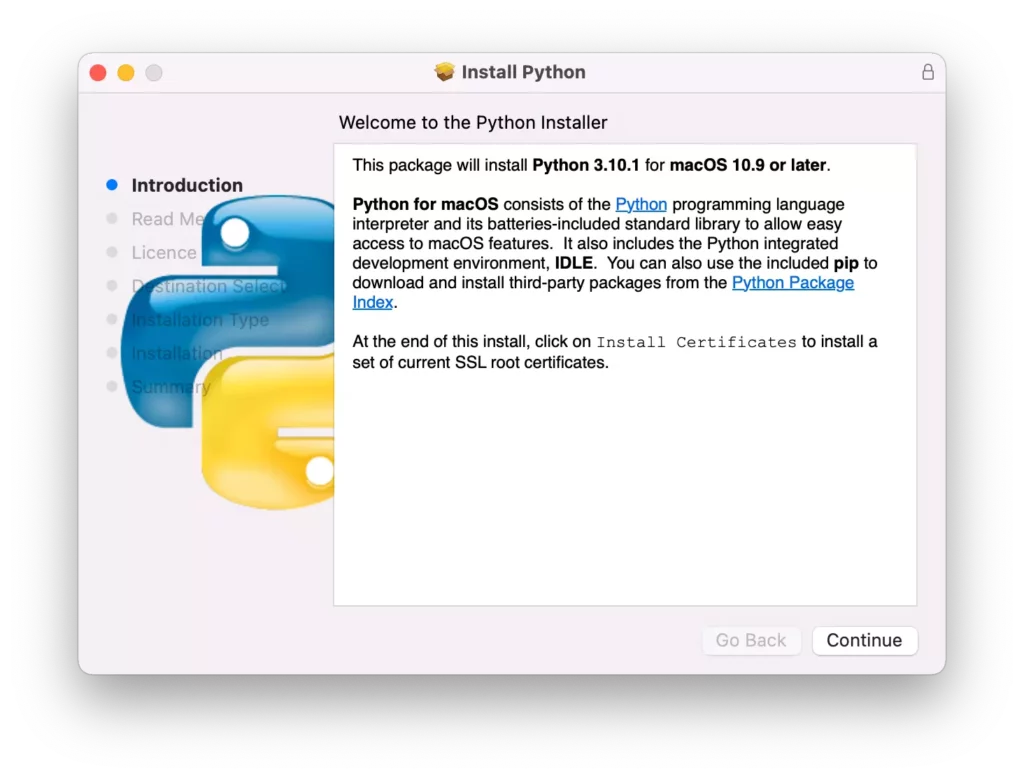


Рис. 3.1.1 – Завантаження Python 3.

Потім завантажив Visual Studio Code.

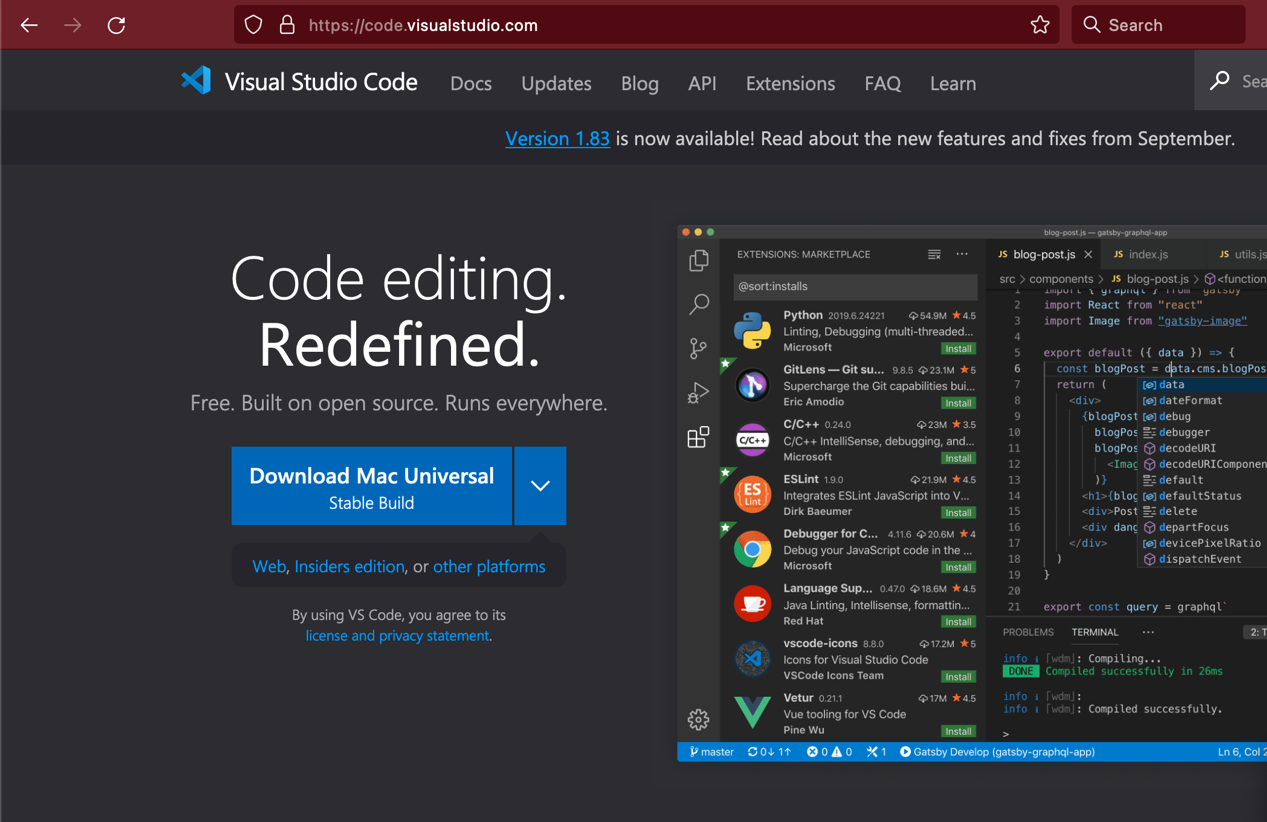


Рис. 3.1.2 – Завантаження Visual Studio Code.

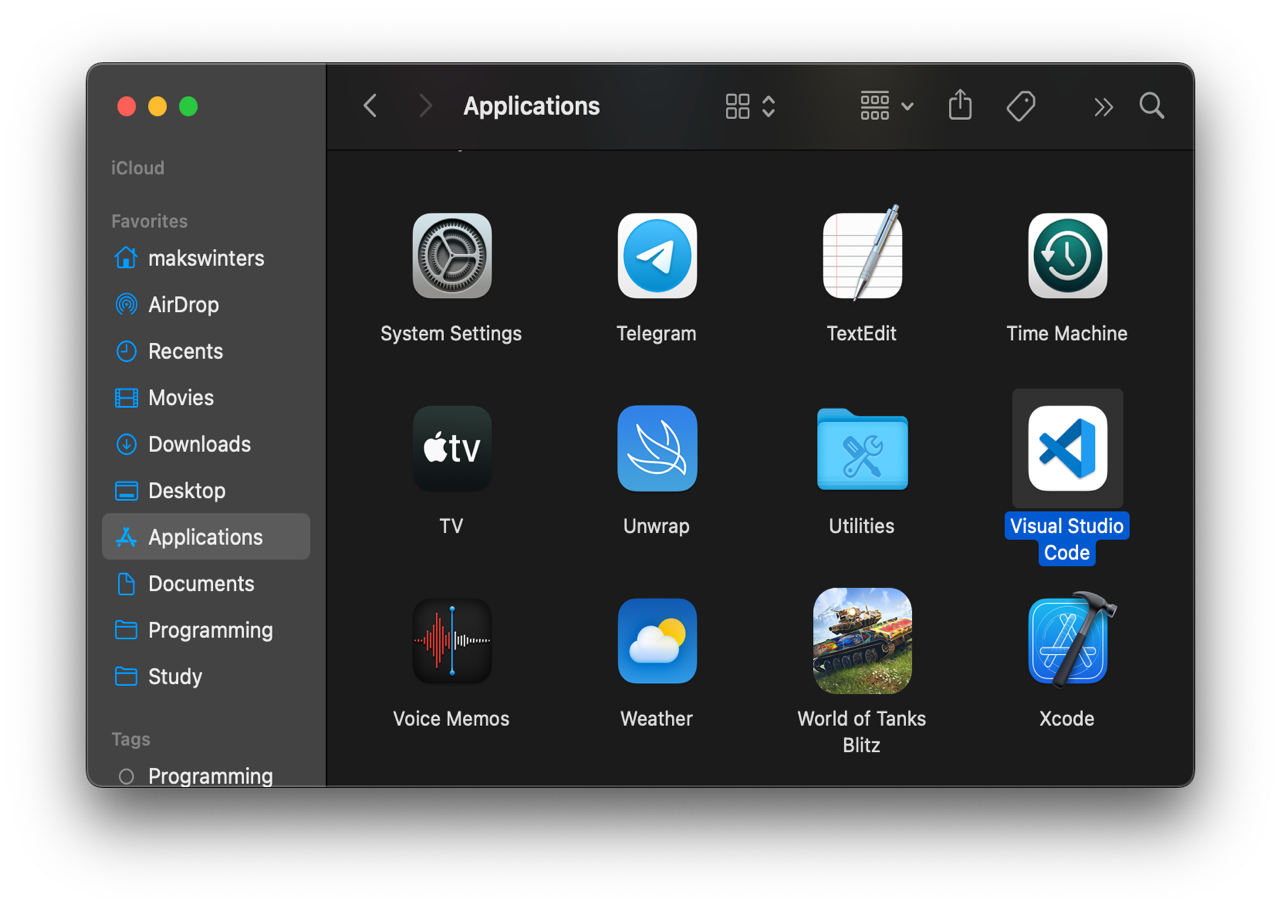
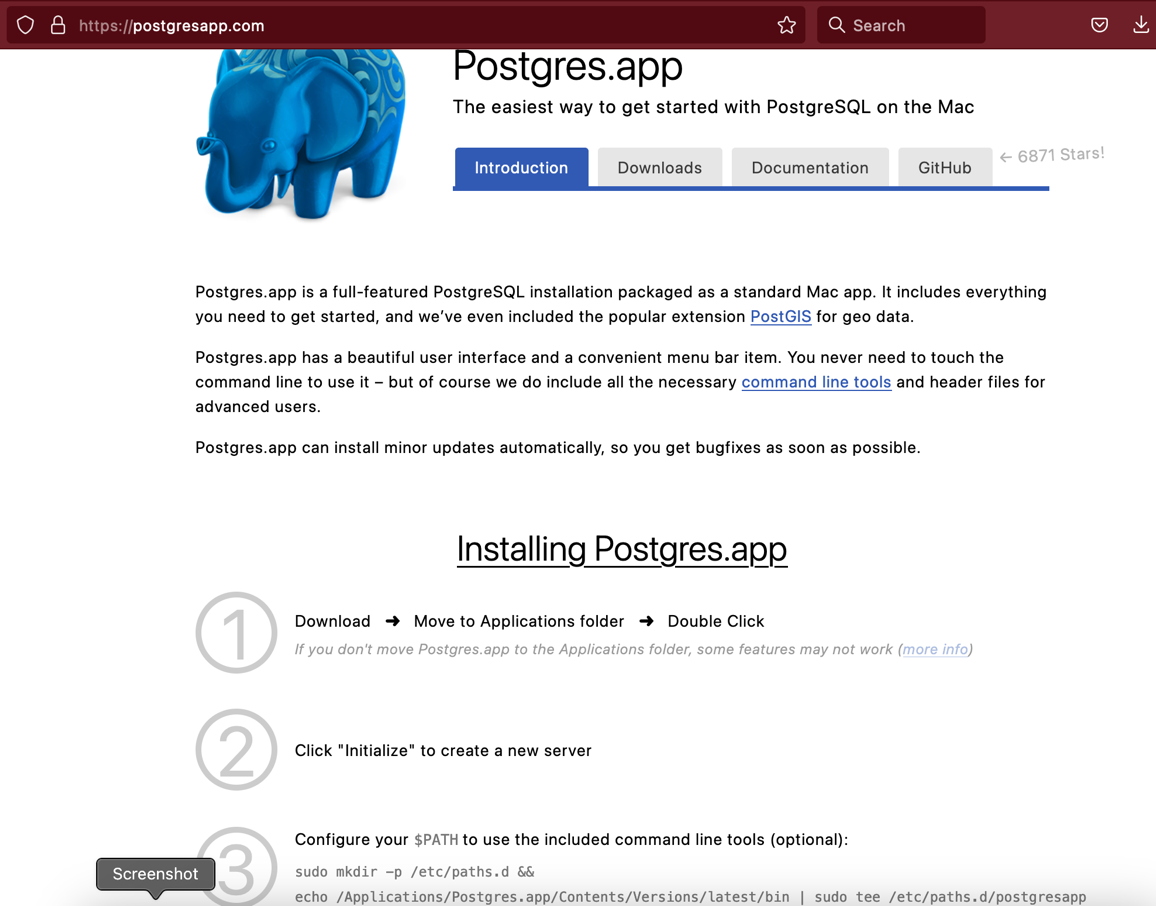


Рис. 3.1.3 – Перетащив программу в папку з додатками.

Рис. 3.1.4 – Встановлення Postgres за iнструкцiею на сайтi.

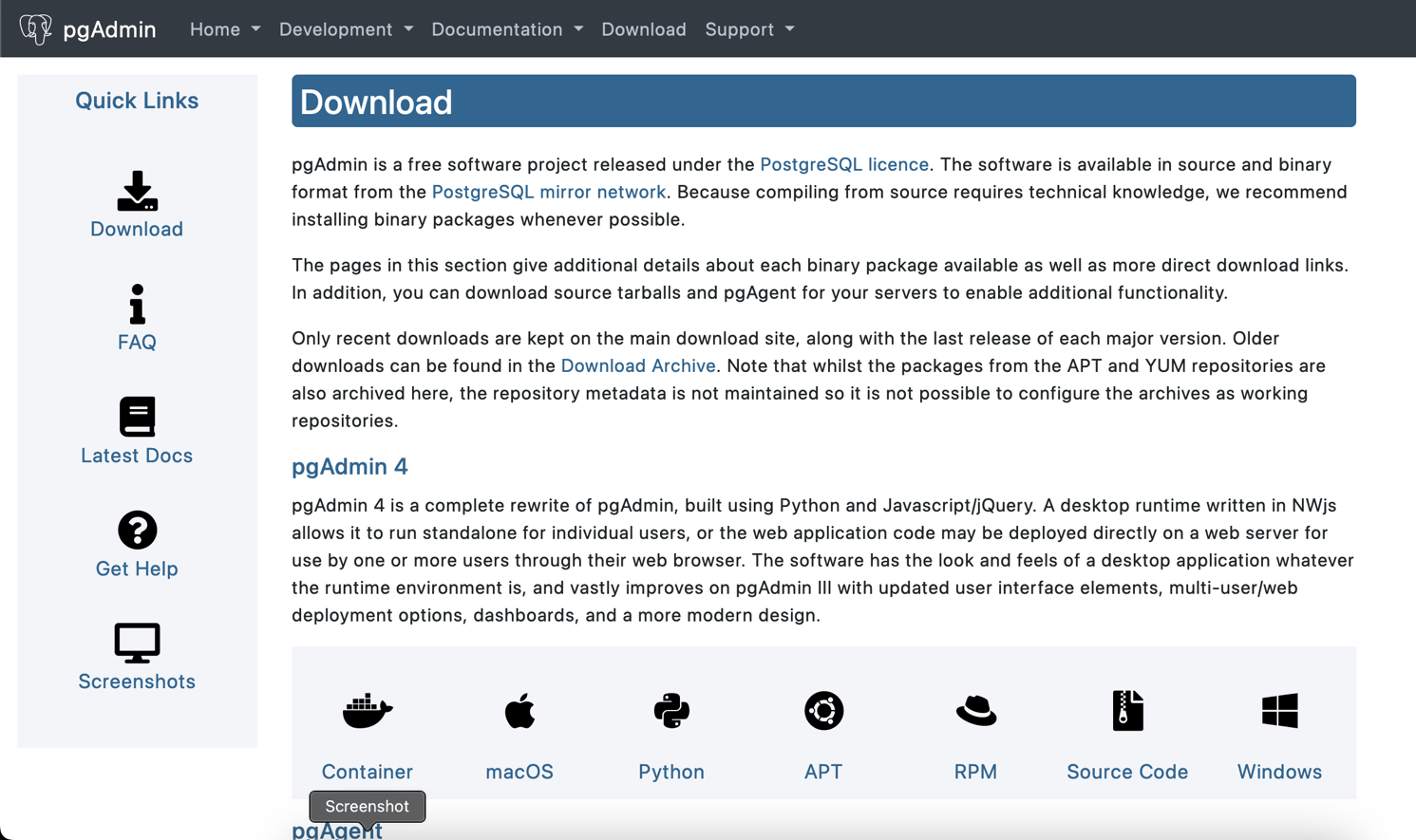


Рис. 3.1.5 – Сайт встановлення pgAdmin.

## 3.2 Робота з мовою програмування Python, бiблiотекою Flask, базою даних Postgres та Flutter Framework.

Python – це універсальна мова, що широко використовується в усьому світі для самих різних цілей – бази даних і оброблення текстів, вбудовування інтерпретатора в ігри, програмування GUI, швидке створення прототипів.

На проектi менi дiсталося завдання створити API для гри на Python використовуючи бiблiотеку Flask.

Написав програму, що викликае потрiбнi функцiї з гри на Python i вiдправляе їх у виглядi JSON (рис. 3.2.1).

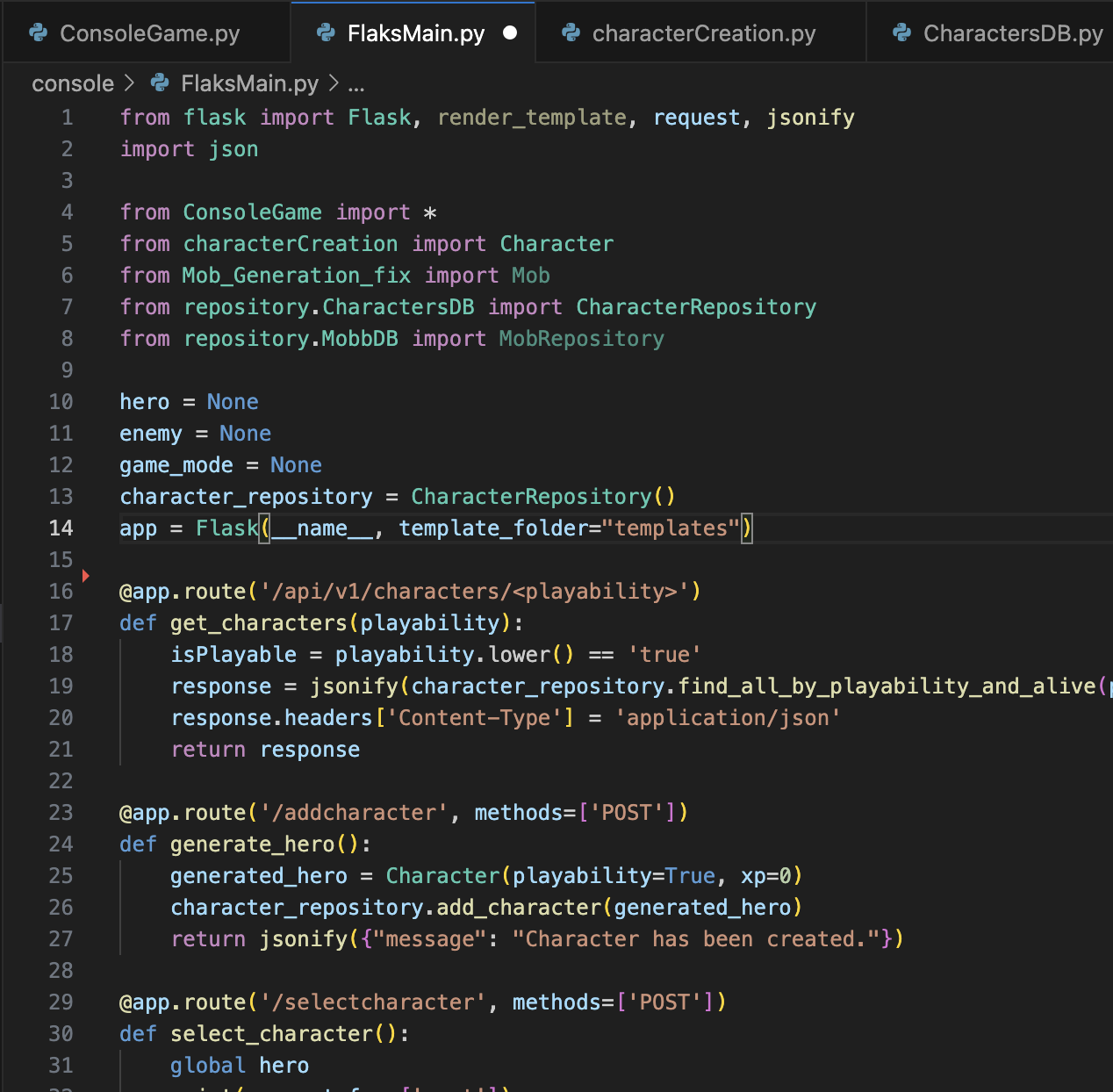
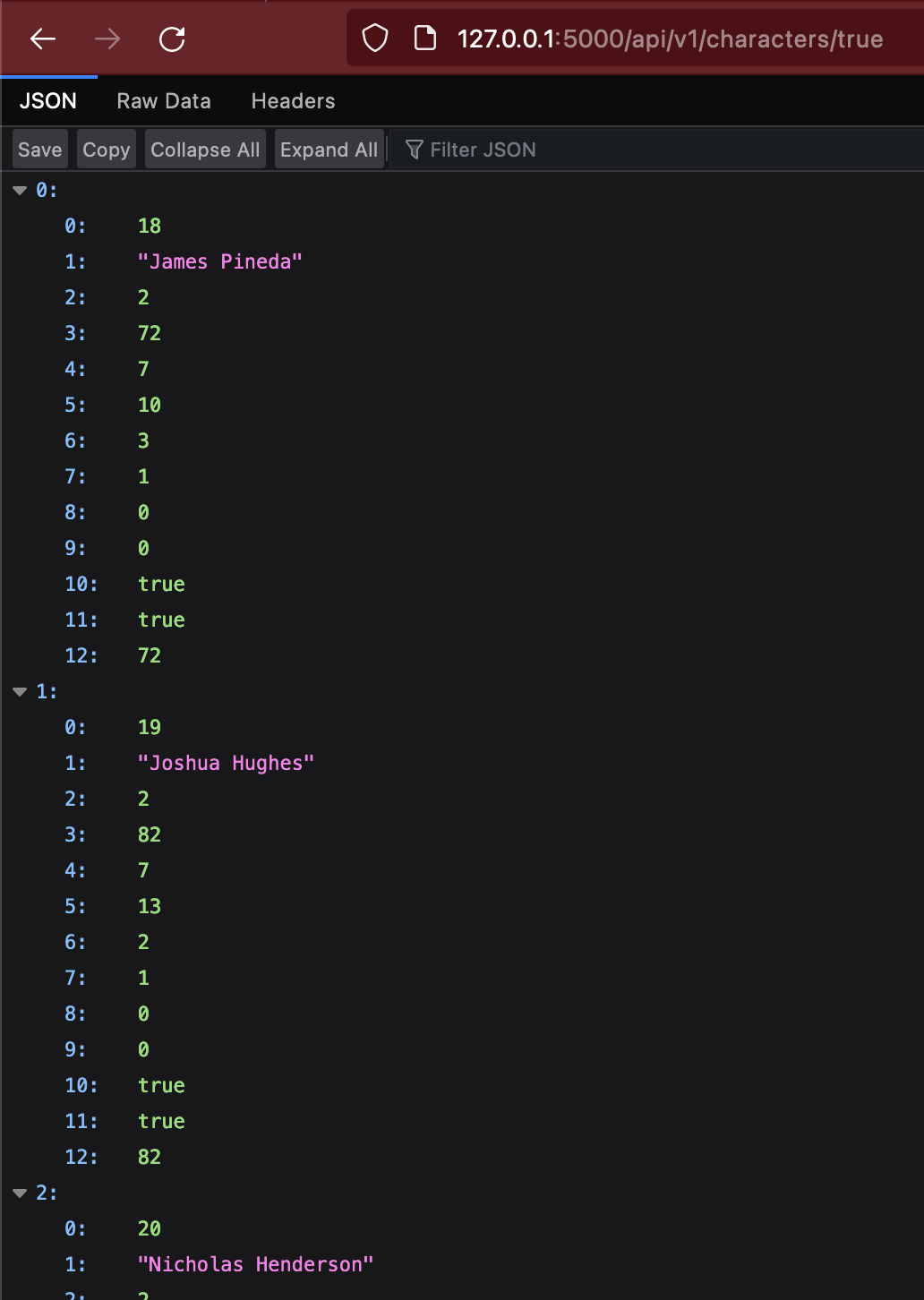


Рис. 3.2.1 – Частина коду програми.

Рис 3.2.2 – Результат роботи API на Flask.

**3.2.1 Робота Postgres**

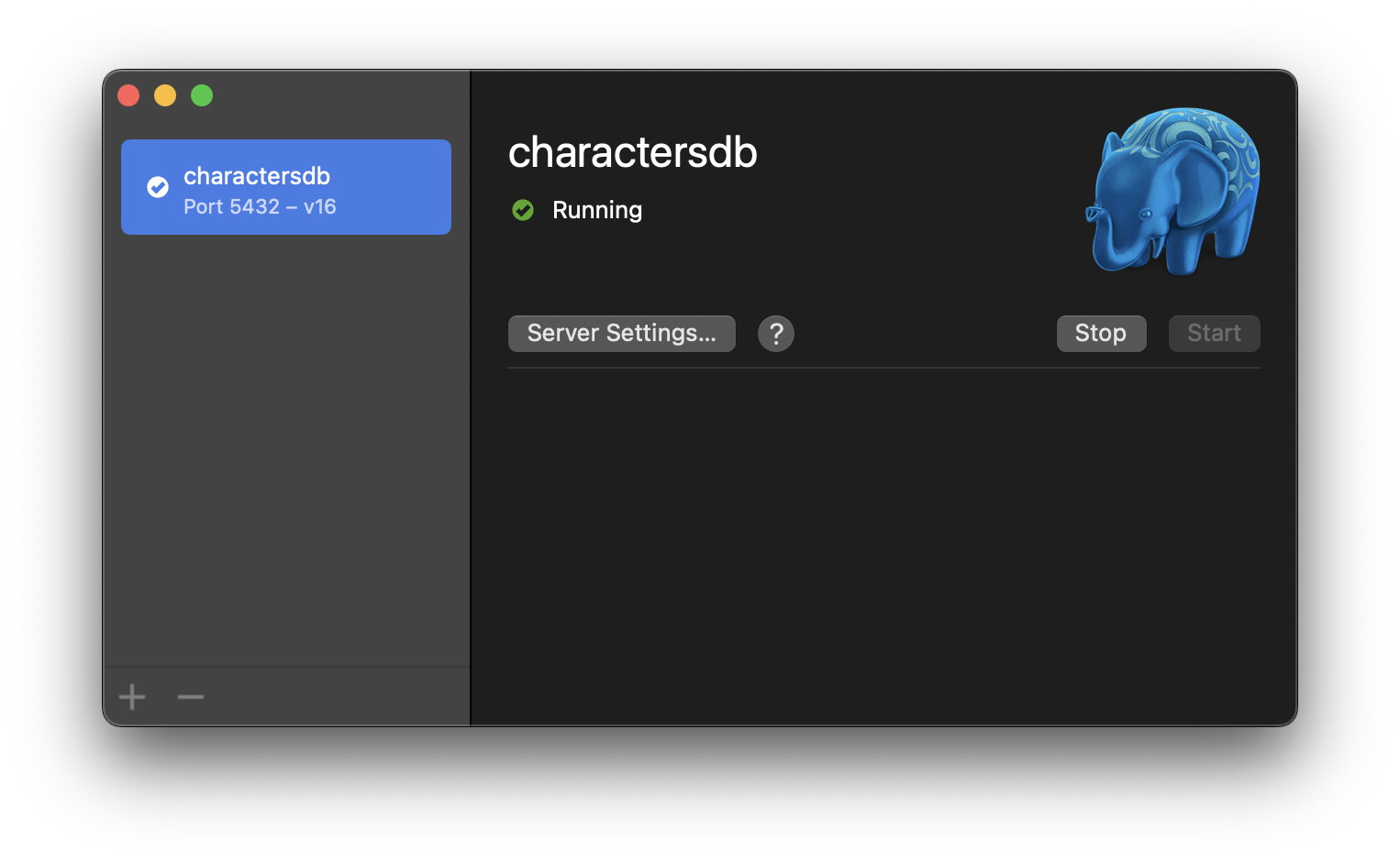
Щоб гра на Python корректно працювала та зберiгала персонажiв її треба з’еднати з базою даних Postgres, для цього вiдкриваю додаток Postgres та запускаю на ньому локальний сервер (рис. 3.2.1.2). 

Рис. 3.2.1.1 – Запуск локального сервера PostgresSQL.

Далi через pgAdmin 4 пiдключаюсь до сервера, це менi знадобиться для роботи з базою даних через графiчний iнтерфейс (рис. 3.2.1.2).

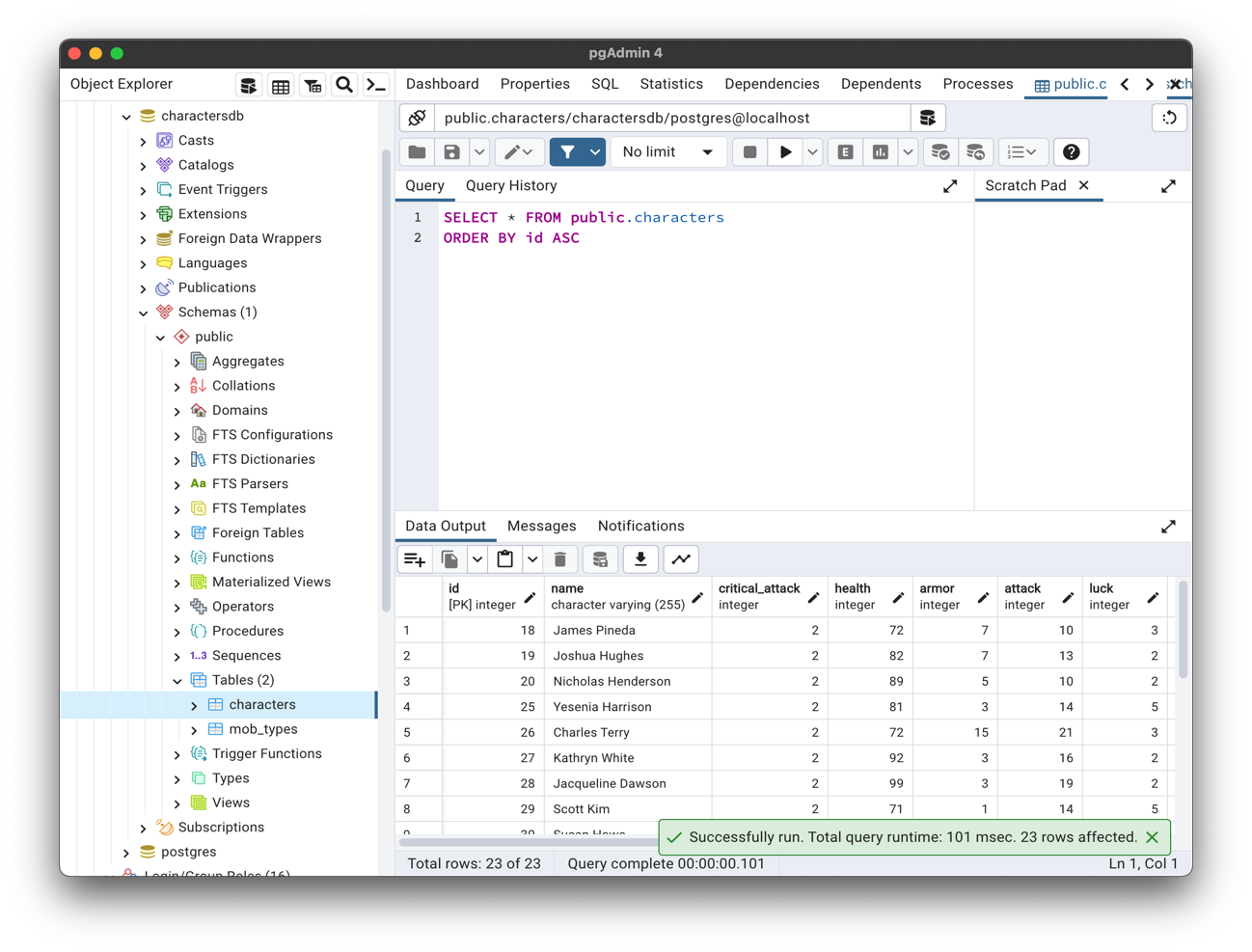


Рис. 3.2.1.2 – Таблиця characters створена грою на Python в pgAdmin 4.

Створив базу данних за допомогою коду в iнструментi query та вiдповiднi таблицi (рис. 3.2.1.3).

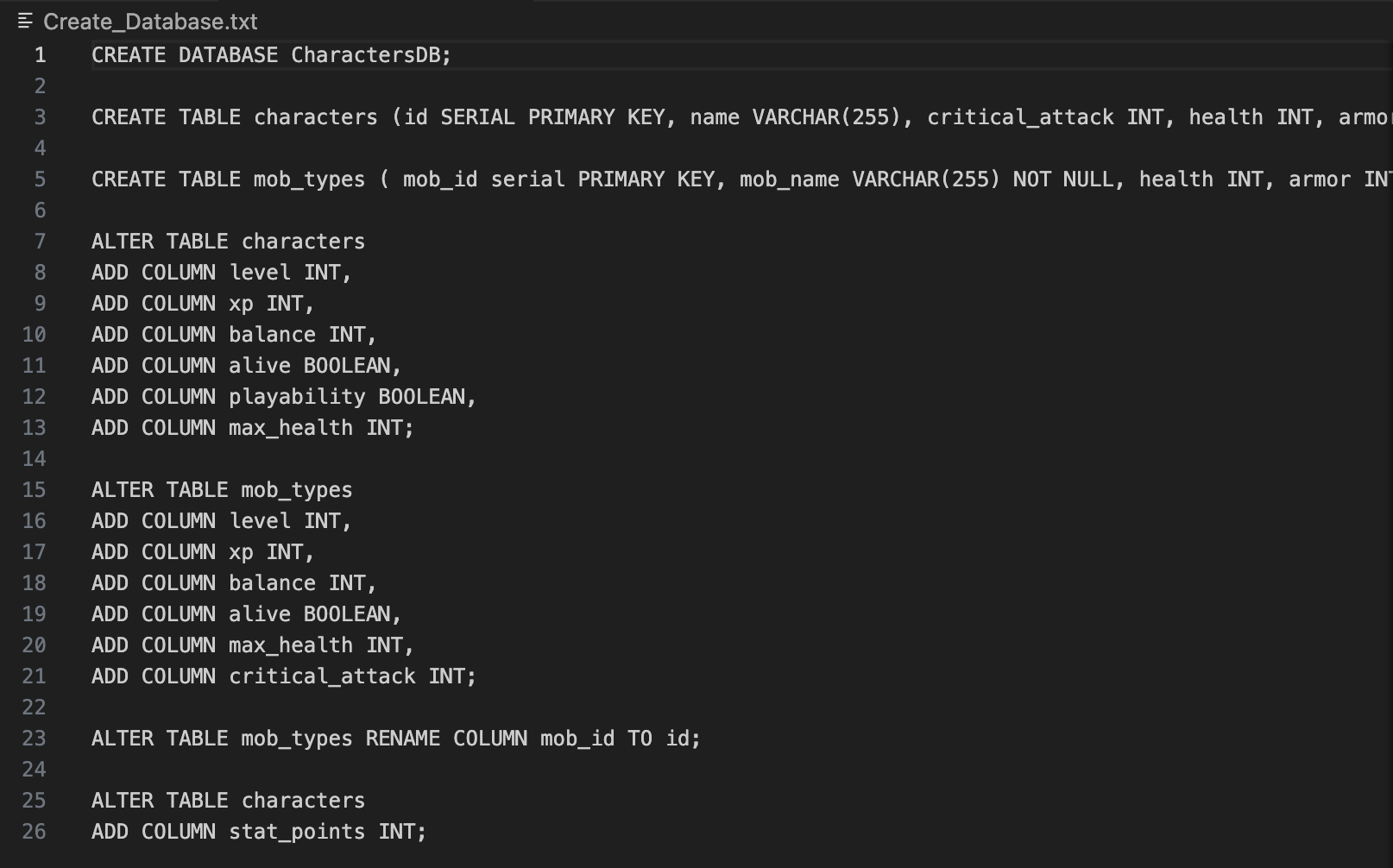


Рис. 3.2.1.3 – Файл створення бази даних.

Далi в класi СharacterRepository в кодi Python потрiбно прописати вiдповiднi данi для пiдключення до серверу бази даних (рис. 3.2.1.4):

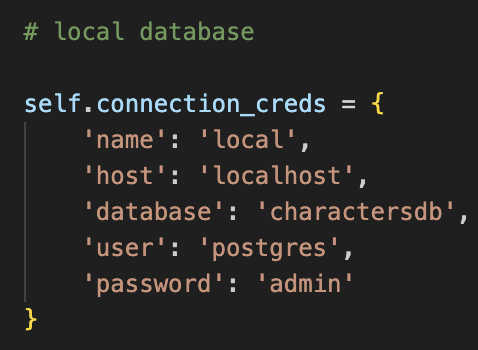


Рис. 3.2.1.4 – Налаштування пiдключення до локальної бази даних в кодi Python.

Пiсля налаштування можна запускати локальний сервер Flask i тестувати його через браузер.

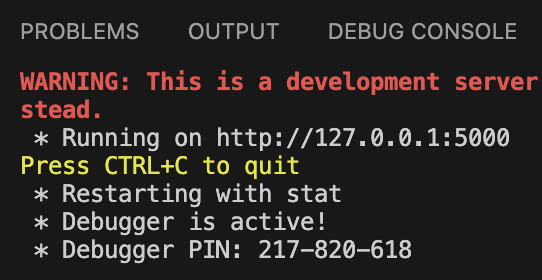


Рис. 3.2.1.5 – Консоль Flask.

### 3.2.2 Робота з Flutter Framework.

Щоб почати працювати з цим фреймворком, потрібно його завантажити.

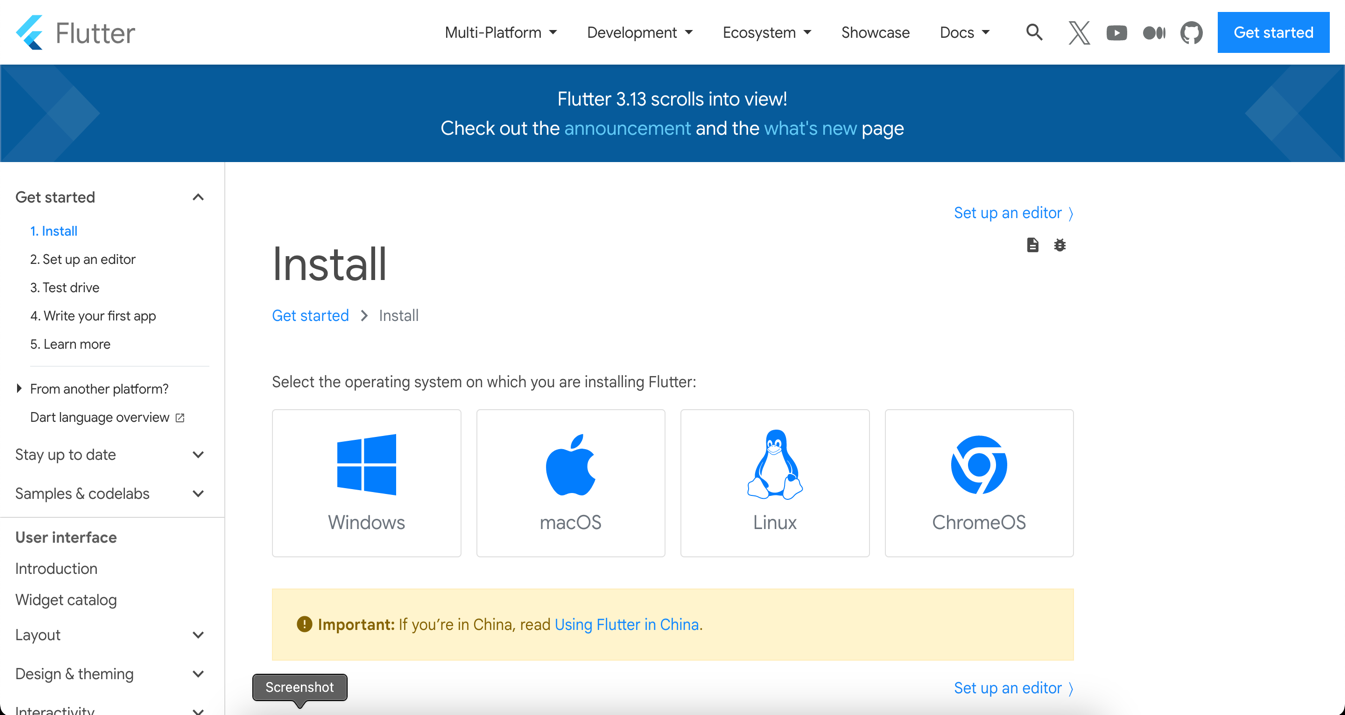
Для цього зайшов на офiцiйний сайт <https://docs.flutter.dev/get-started/install> та обрав потрiбну менi ОС (рис. 3.2.2.1)

Рис. 3.2.2.1 – Сторiнка завантаження Flutter Framework.

Розпакував у зручний шлях та ввiв команду для Path (рис. 3.2.2.2):

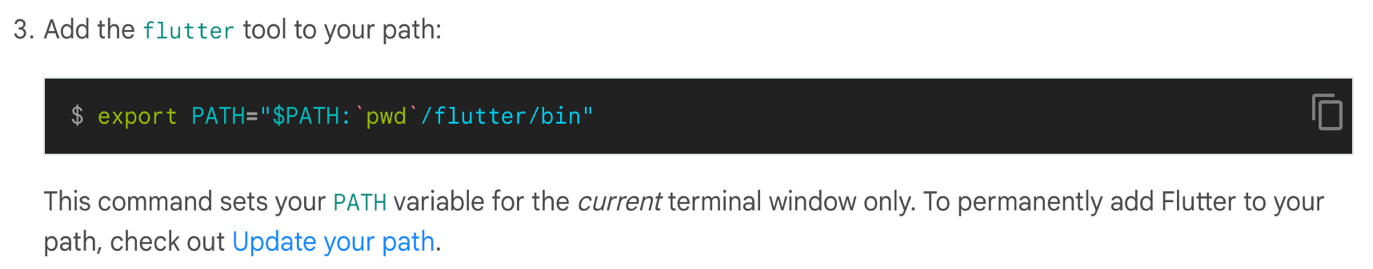


Рис. 3.2.2.2 – Додавання Flutter в PATH системи.

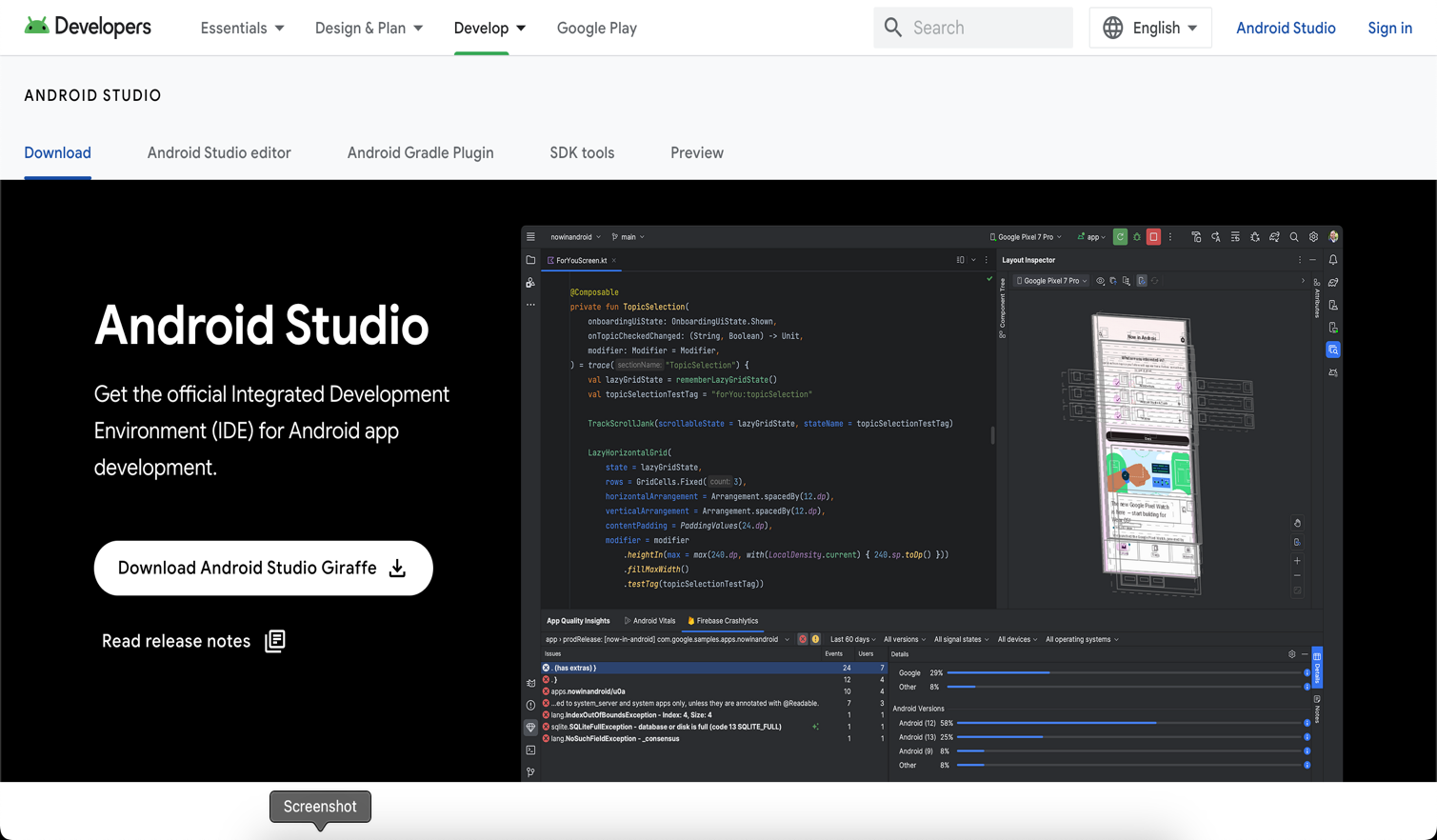
Для користування та будування додаткiв пiд Android потрiбно завантажити Android Studio який автоматично завантажить все потрiбне (рис 3.2.2.3):

Рис. 3.2.2.3 – Cторiнка завантаження Android Studio.

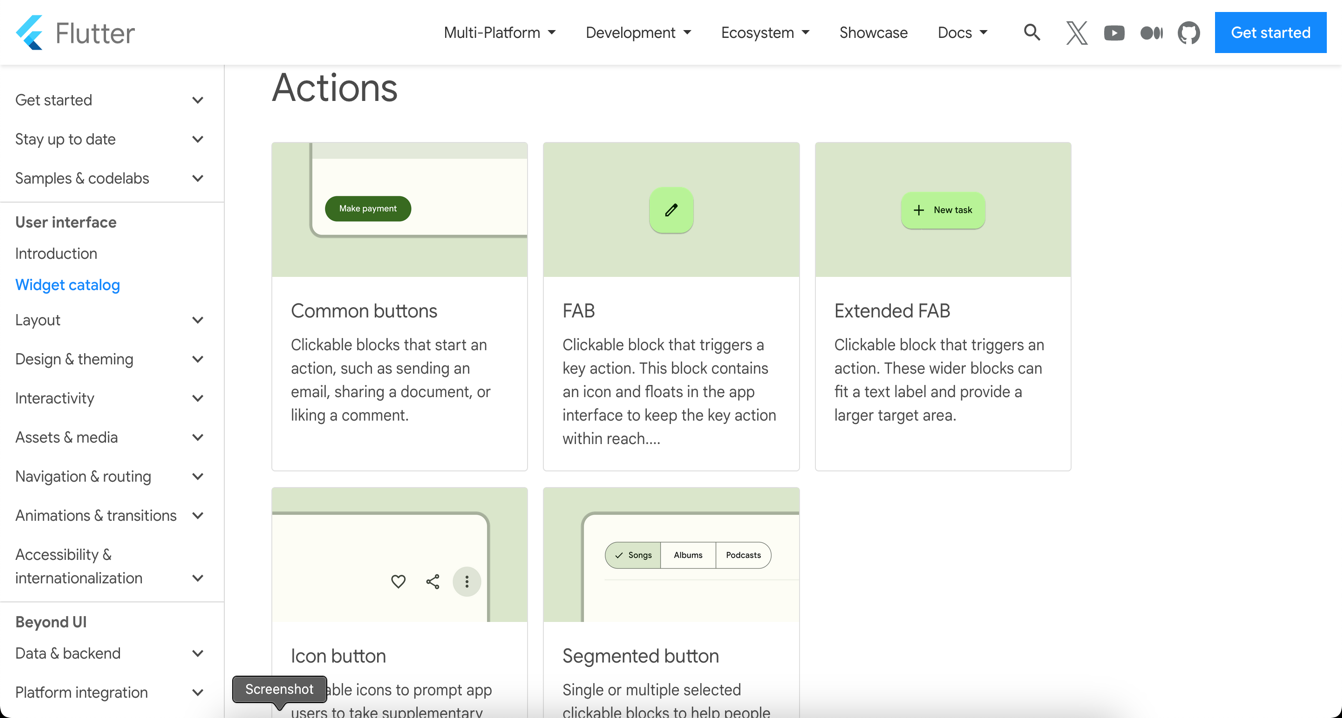
Перейшов на сторiнку <https://docs.flutter.dev/ui/widgets/material> (Material Components Widgets), де можна побачити приклади iмплементацiї стандартних вiджетiв в Flutter (рис 3.2.2.4).

Рис. 3.2.2.4 – Стандартнi компоненти/вiджети Flutter.

За допомогою цих компонентів Flutter я створив інтерфейс для нашого консольного застосунку, присвяченого бою персонажів. Наш додаток включає в себе кілька ключових екранів, які допомагають користувачеві насолоджуватися грою (рис 3.2.2.5 – 3.3.2.9).

1. **Екран вибору персонажа:** На цьому екрані гравець може обрати персонажа, за якого він хоче грати. Використовуючи Flutter, ми створили інтерактивний інтерфейс, який дозволяє користувачеві прокручувати список доступних персонажів, переглядати їхні зображення та властивості, а також вибирати свого улюбленого бійця.
2. **Екран вибору ігрового режиму:** Наш додаток пропонує кілька режимів гри, і цей екран допомагає гравцеві обрати, який режим його цікавить. Використовуючи Flutter, ми створили кнопки або вкладки, які представляють різні режими, і користувач може з легкістю вибрати потрібний режим гри.
3. **Екран вибору супротивника:** Перед початком бою гравець повинен вибрати супротивника. На цьому екрані ми використовуємо Flutter, щоб створити інтерфейс для пошуку. Ми показуємо зображення супротивників та їхні статистичні дані, щоб гравцю було легше обрати суперника.
4. **Екран бойовки:** Цей екран є серцем нашого додатка, де відбувається сама битва. Використовуючи Flutter, ми створили інтерактивний ігровий інтерфейс, який включає в себе зображення персонажів, відображення життєвих показників, кнопки для атаки, захисту та інших ігрових дій. Цей екран може бути як статичним, так і анімованим, щоб зробити гру більш захоплюючою.Top of Form

Bottom of Form



Рис. 3.2.2.5 – Екран вибору персонажа для гри.



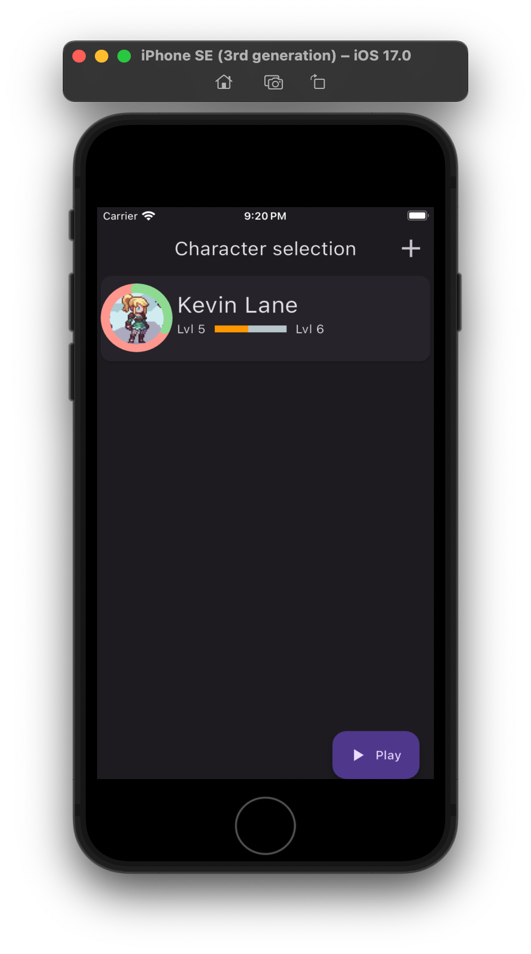
Рис. 3.2.2.6 – Екран вибору iгрового режиму.

Рис. 3.2.2.7 – Екран перегляду списку ворогiв.

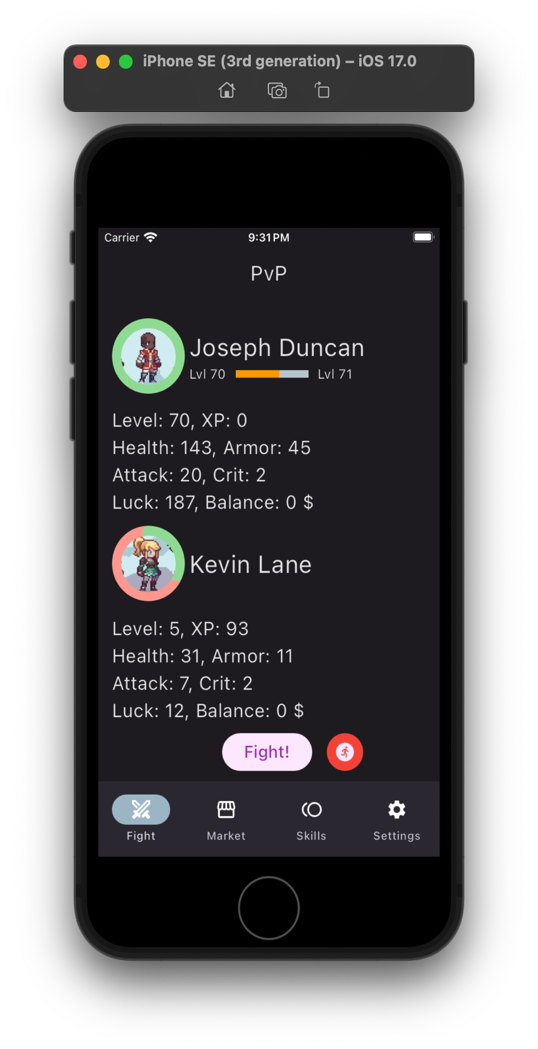


Рис. 3.2.2.8 – Екран битви.

Рис. 3.2.2.9 – Екран перемоги.

### 3.2.3 Робота з Git та GitHub

Git - це система керування версіями, яка дозволяє розробникам відстежувати та керувати змінами в програмному коді проекту. Вона дозволяє зберігати всі версії файлів і легко порівнювати їх між собою, створювати гілки для одночасної роботи над різними функціями та об'єднувати зміни в одну основну версію. Git допомагає уникнути конфліктів при роботі з командою та дозволяє ефективно керувати проектом.

GitHub, з іншого боку, є хостинговою платформою для Git-репозиторіїв. Він надає зручний спосіб завантажувати Git-репозиторії в хмару, де вони стають доступними для спільної роботи та співпраці з іншими розробниками. GitHub також надає інструменти для відстеження проблем, зберігання документації та аналізу історії змін проекту.

Як студент, я використовував Git та GitHub під час розробки проекту гри битви персонажів. За допомогою Git, ми зберігали код гри та всі зміни в ньому, створюючи нові гілки для різних функцій гри. Це дозволяло нам працювати над різними аспектами гри паралельно, не перешкоджаючи один одному. Коли робота над функцією була завершена, ми зливали гілку з цією функцією у головну гілку проекту, щоб інтегрувати її зміни.

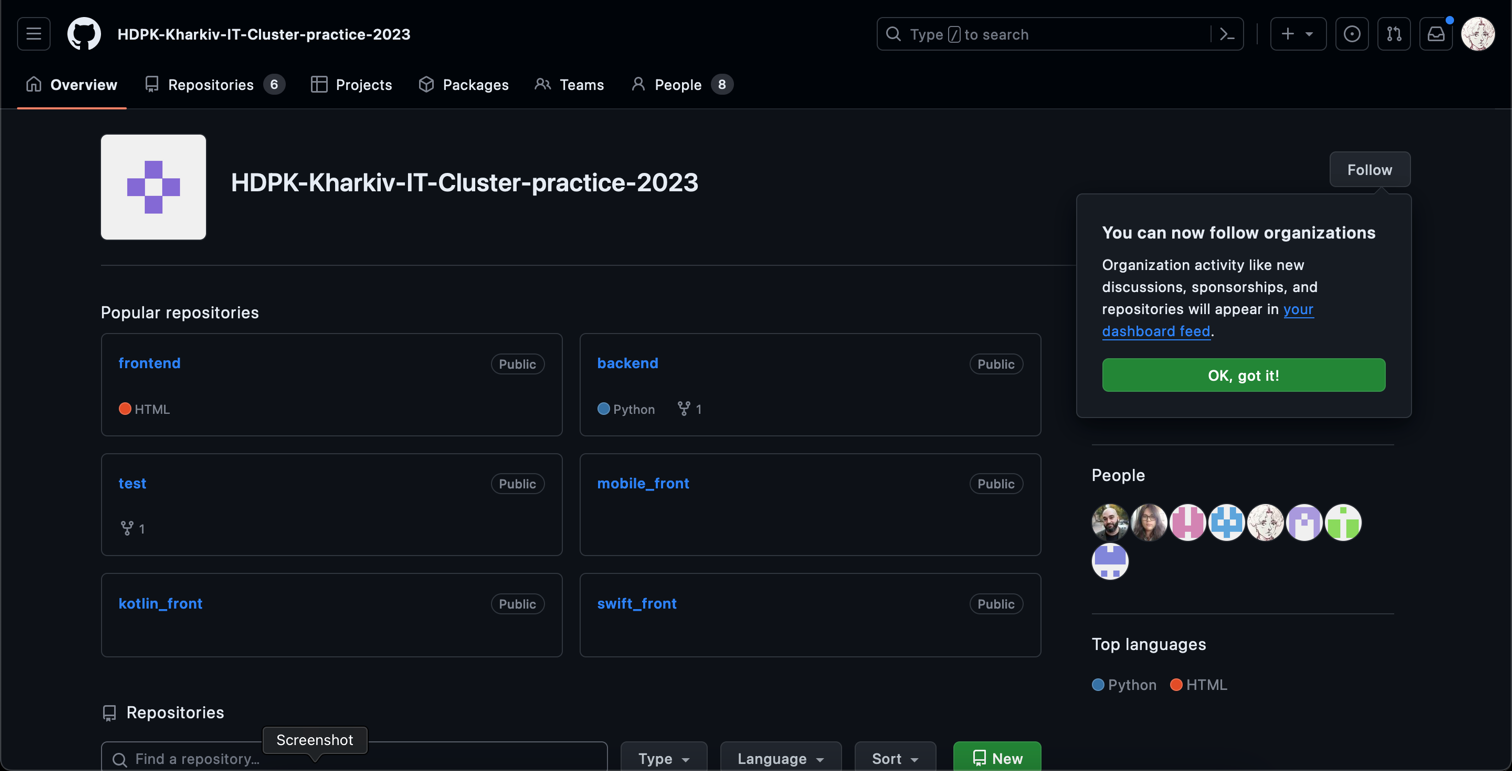
GitHub надавав нам можливість завантажувати наш проект, створювати проблеми для відстеження завдань та відгалужувати проект для співпраці з іншими розробниками. Це спрощувало комунікацію та спільну роботу над грою. GitHub також був корисний для зберігання документації та аналізу наших змін, що допомагало нам створити звіт про проект.

Рис. 3.2.3.1 – Головна сторiнка нашої органiзацiї на GitHub.

На проекті є різні репозиторії, для прикладу візьмемо mobile\_front. Репозиторій - це централізоване сховище для зберігання і керування версіями файлів і програмного коду. У вас є можливість працювати з цим репозиторієм і виконувати різні дії, які сприяють спільному розробленню та управлінню проектом. Ось деякі з них:

1. **Клонування (Clone)**: Ви можете створити копію репозиторію на своєму комп'ютері за допомогою команди git clone. Це дозволяє вам мати локальну копію проекту для редагування та тестування.
2. **Зміни (Commit)**: Ви можете вносити зміни в файли проекту та фіксувати їх за допомогою команди git commit. Кожен коміт представляє собою "знімок" змін, які ви внесли, і має опис того, що було змінено.
3. **Гілки (Branches)**: Ви можете створювати гілки (branches), що є окремими шляхами розвитку проекту. Це дозволяє вам працювати над різними функціями або виправленнями паралельно без впливу на головну гілку.
4. **Об'єднання (Merge)**: Після того, як ви внесли зміни в гілці, ви можете їх об'єднати з головною гілкою за допомогою команди git merge.
5. **Завантаження (Push)**: Коли ви готові поділитися своїми змінами з іншими учасниками проекту, ви можете завантажити їх на сервер репозиторію за допомогою команди git push.
6. **Оновлення (Pull)**: Ви можете отримати останні зміни з сервера, щоб мати актуальну версію проекту, використовуючи команду git pull.
7. **Виправлення конфліктів (Conflict Resolution)**: Якщо інші користувачі внесли зміни в ті ж файли, які ви редагували, можуть виникнути конфлікти. Ви можете їх вирішити вручну, щоб об'єднати зміни.
8. **Історія (History)**: Ви можете переглядати історію комітів, щоб дізнатися, хто і коли вносив зміни в проект.

Загалом, репозиторій - це потужний інструмент для спільної роботи над проектами, що дозволяє вам ефективно вести керівництво версіями, контролювати доступ і зберігати історію змін.

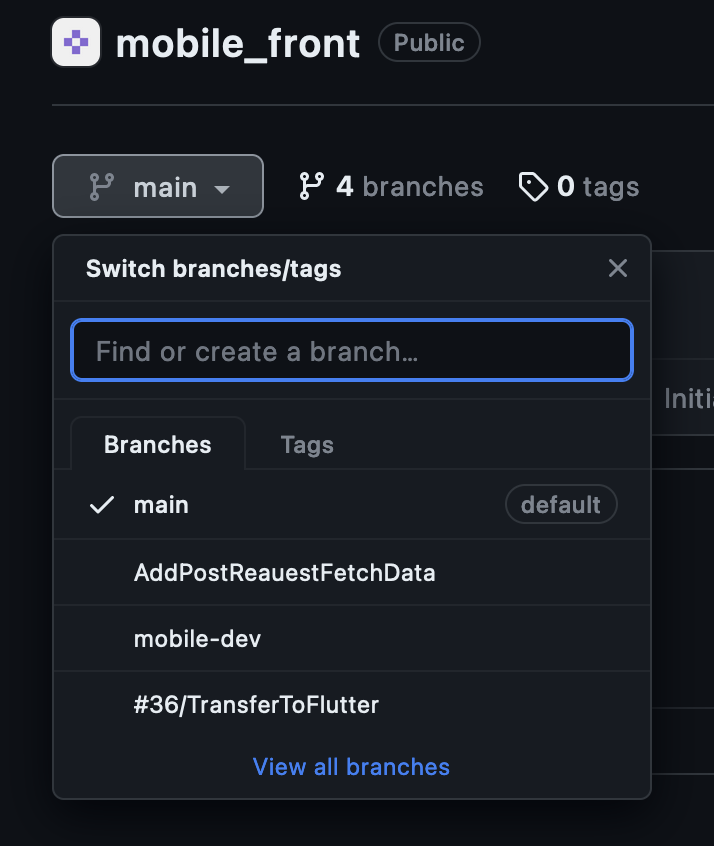


Рис. 3.2.3.2 – Гiлки рипозиторiю mobile\_front.

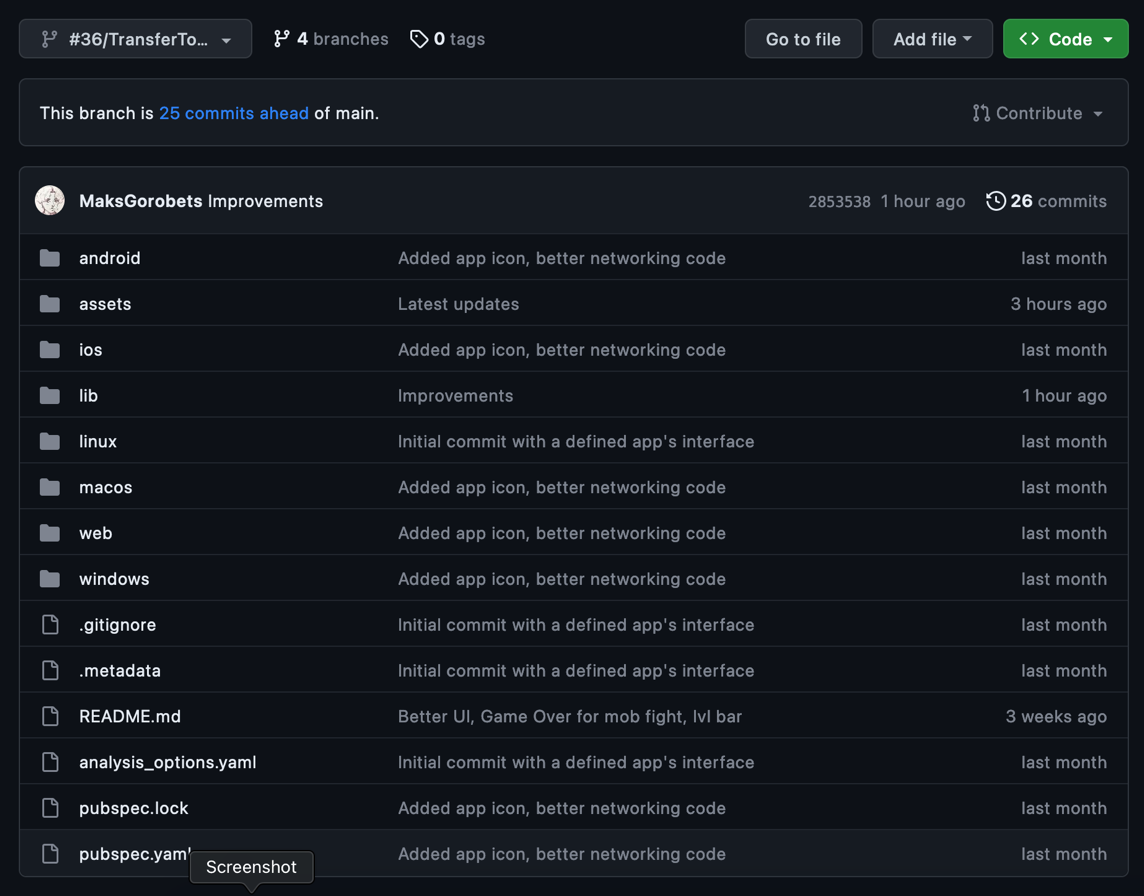


Рис. 3.2.3.3 – Файли проекту на гiлцi №36/TransferToFlask.

Для роботи з GitHub на ПК повинен бути встановленний Git, у випадку з Mac OS вiн йде у комплектi з ОС.

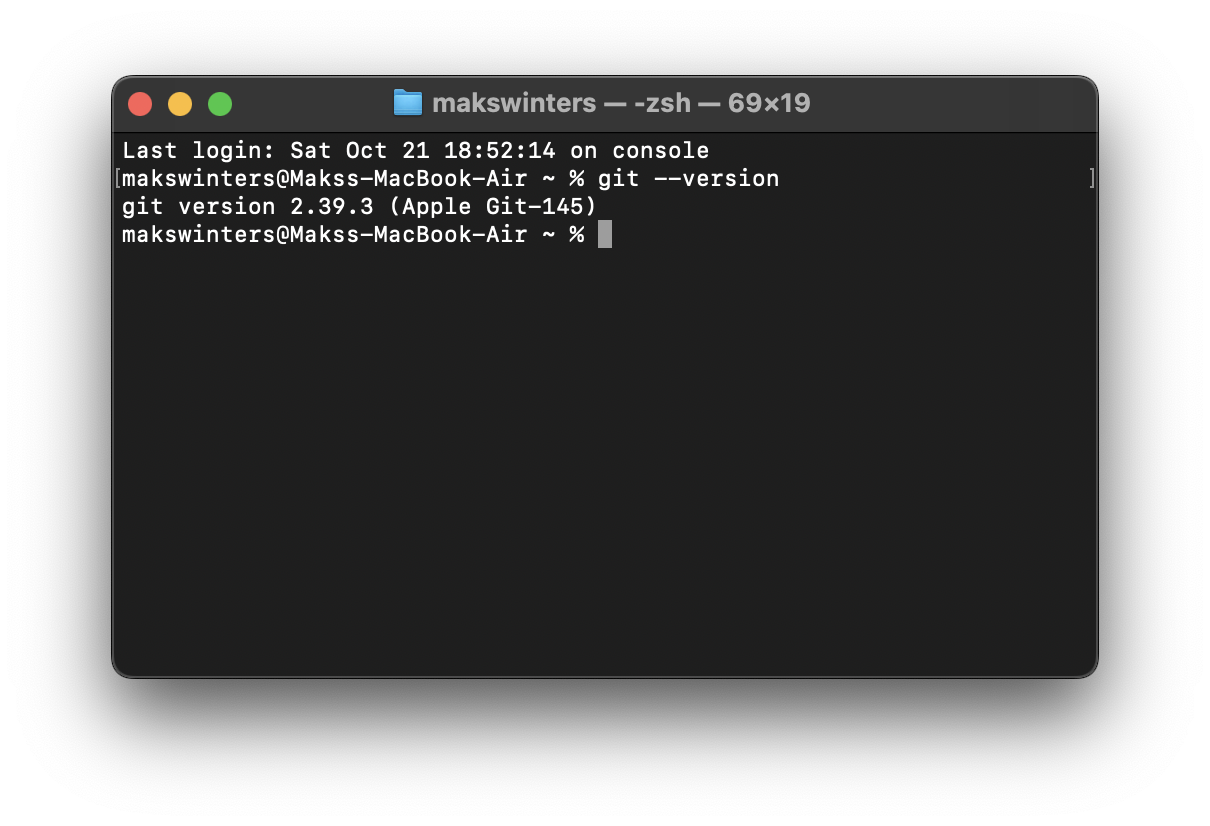


Рис. 3.2.3.4 – Результат команди git –version.

Для клонування репозиторію з GitHub використовуються команди Git. Ось послідовність команд для клонування репозиторію і пояснення, чому це потрібно робити:

1. **git clone**: Ця команда створює локальну копію віддаленого репозиторію на вашому комп'ютері. Синтаксис команди:

git clone <URL репозиторію>

Наприклад, якщо ви хочете клонувати репозиторій з GitHub, то URL може виглядати, наприклад, так:

git clone https://github.com/username/repo.git

Пояснення: Клонування репозиторію з GitHub дозволяє вам мати локальну копію проекту на своєму комп'ютері. Це особливо корисно для роботи над кодом або файлами, оскільки ви можете вносити зміни, створювати гілки, тестувати функціональність та зберігати історію змін локально. Крім того, це дозволяє вам працювати в автономному режимі, навіть коли ви не підключені до Інтернету.

Пicля клонування репозиторiю папку можна вiдкрити у редакторi коду VSCode та данi про нього автоматично з’являться у програмi. (рис. 3.2.3.5)

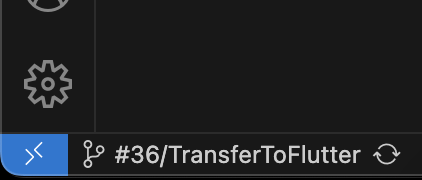


Рис. 3.2.3.5 – Гiлка рипозиторiю mobile\_front в VSCode.

Пicля роботи над кодом роблю commit щоб зафiксувати змiни а потiм push для того щоб цi змiни також вiдобразилися на GitHub.

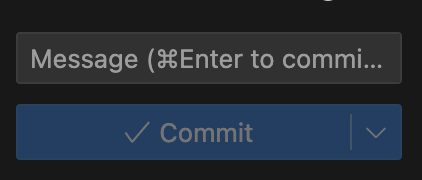


Рис. 3.2.3.6 – Вiконце введення повiдомлення для сommit.

# **ВИСНОВОК**

Під час моєї практики я отримав нові знання та навички. Конкретно, я поглибив своє розуміння мови програмування Python та вивчив бібліотеку Flask, ознайомившись із принципами її роботи та можливостями створення веб-сторінок.

Також я опанував основи створення мобільних додатків, використовуючи Flutter framework та мову програмування Dart. У процесі цього навчання я здобув досвід у роботі з нетворкінгом в мобільних додатках, включаючи GET та POST запити, роботу з форматом JSON та використання сторонніх бібліотек Flutter.

# **ПЕРЕЛІК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ**

1. Flutter Apprentice (Third Edition): Learn to Build Cross-Platform Apps

URL: <https://www.amazon.com/Flutter-Apprentice-Third-Learn-Cross-Platform/dp/1950325741>

1. Python Crash Course, 3rd Edition: A Hands-On, Project-Based Introduction to Programming

URL: <https://www.amazon.com/Python-Crash-Course-Eric-Matthes/dp/1718502702>

1. Swaroop CH. A Byte of Python. 2022.

URL: [https://github.com/swaroopch/byte-ofpython/releases/tag/v239134197cc453397b0540fa051392af6b47f9f3/byteof-python.pdf](https://github.com/swaroopch/byte-of-python/releases/tag/v239134197cc453397b0540fa051392af6b47f9f3/byte-of-python.pdf)

1. Pro Git Book 2nd edition (2014)

URL: <https://git-scm.com/book/en/v2>