

## «РОЗРОБКА ОРГАНАЙЗЕРУ TO-DO-LIST»

*І. О. Сенік, П. В. Римар*

*Анотація.* Стаття присвячена розробці програмного продукту для полегшення планування власних справ за ступенями важливості та терміновості. В роботі наведена переваги та недоліки даного програмного продукту. Практична частина розробки – десктоп – додаток, який розроблений для платформи Windows.

*Ключові слова:* Python, десктоп – додаток, програмний продукт, Матриця Ейзенхауера.

**Вступ.** Кожна людина на сьогоднішній день має досить багато справ, які забирають у неї великий відсоток часу. Щоб не загубитися в потоку цих справ, які ллються рікою, необхідно мати спеціальне програмне забезпечення для полегшення слідкування за власними справами. Зникає необхідність постійно купувати дорогі блокноти та носити їх з собою у сумці, листки А4 також не безкінечні і також не зручні. Відповідно, це також значна економія коштів. Тому краще записувати свої справи у спеціальній програмі, в якій можна їх розподіляти по ступеню важливості.

Мета роботи полягає у вивченні методики проектування десктоп-додатків на мові програмування Python.

Задачами дослідження є:

1. Огляд існуючих програм за даною тематикою, виявлення їх переваг та недоліків.

2. Ознайомленні та вивчені спеціалізованого середовища розробки (JetBrains) для проектування десктоп-додатку.

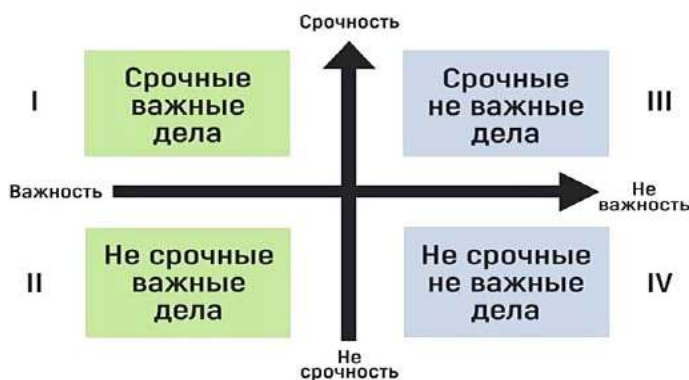
Об'єктом для дослідження є мова програмування «Python», а саме, наскільки зручно на її основі проектувати десктоп-додатки.

Предметом для дослідження є механізм розробки органайзера для власних справ на базі мови програмування «Python».

В ході роботи буде застосовуватись бібліотека «tkinter» з мови програмування Python. В якості бази даних виступає Microsoft Access. Для проектування використовується архітектурний шаблон «MVC».

**Основна частина.** Логіка десктоп-додатку побудована на основі матриці Ейзенхауера.

Матриця Ейзенхауера – досить відомий та ефективний метод короткострокового планування. Ідея матриці Ейзенхауера полягає у розстановці пріоритетів по ступені важливості та терміновості. Матриця Ейзенхауера складається із 4-х квадратів, які мають різний пріоритет. Кожну заплановану справу у залежності від важливості та терміновості потрібно записати у відповідний квадрат. На рис. 1 наведено розподіл справ по категоріях, який був запропонований Ейзенхауером.



*Рис. 1. Розподіл справ по категоріях*

Всі справи можна віднести до таких категорій:

**I** Невідкладні завдання: проект, в якого наближаються термін здачі; вирішення проблем зі здоров'ям.

**II** Планування нових та аналіз старих проектів; вивчення іноземних мов; відвідування спортзалу.

**III** Громадська діяльність; збори, наради, мітинги; справи, які не стосуються ваших посадових обов'язків.

**IV** Неважливі розмови по телефону; перегляд телебачення; комп'ютерні ігри; перегляд розважальних порталів, спілкування у соціальних мережах.

**Аналоги.** На сьогоднішній день є багато різних органайзерів для запису своїх справ, але на ринку десктоп додатків, які б використовували матрицю Ейзенхауера – їх не має. За аналог буде взято android-додаток, назва якого «4Today», цей продукт схожий на розроблюваний, але відмінність полягає в тому, що він зроблений для смартфонів на платформі Android. На наступному рисунку наведений екран програми з певної категорії.

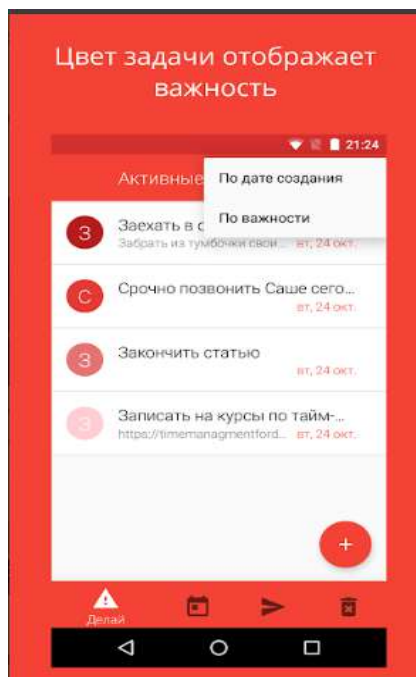


Рис. 2. Органайзер «4Today»

Головний екран розроблюваного додатку містить 4 квадрати з різними кольорами (в залежності від ступеню важливості та терміновості).

1. Червоний – терміново.
2. Зелений – важливо, але не дуже терміново.
3. Жовтий – терміново, але не важливо.
4. Синій – Не терміново, не важливо.

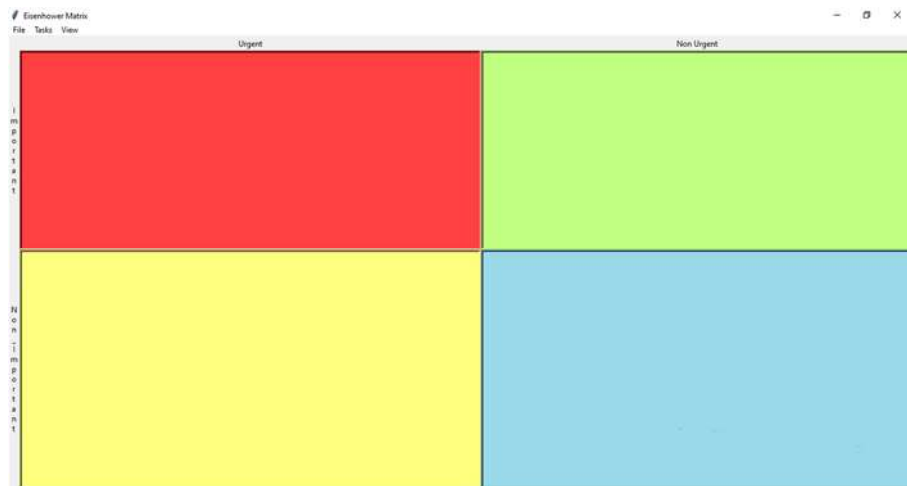
Також можна бачити 2 ступені важливості та 2 ступені терміновості.

Ступені Важливості:

1. Important – важливо.
2. Not Important – не важливо.

Ступені Терміновості:

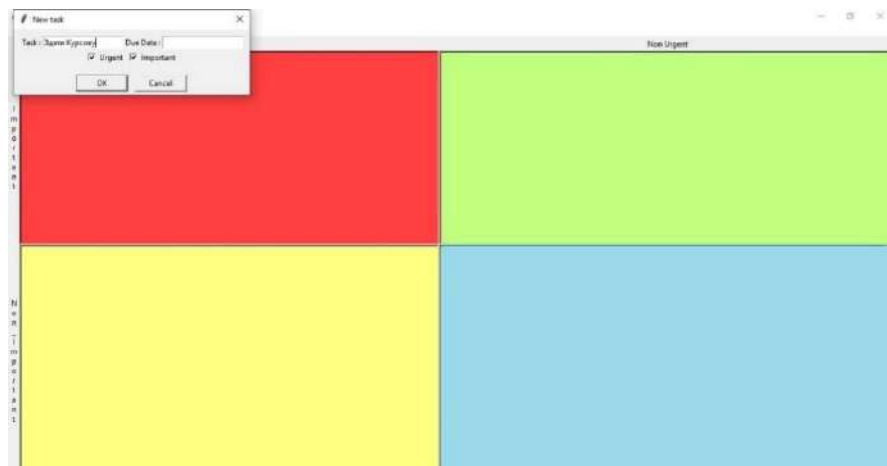
1. Urgent – терміново.
2. Not Urgent – не терміново.



*Рис. 3. Головний екран «Власних справ»*

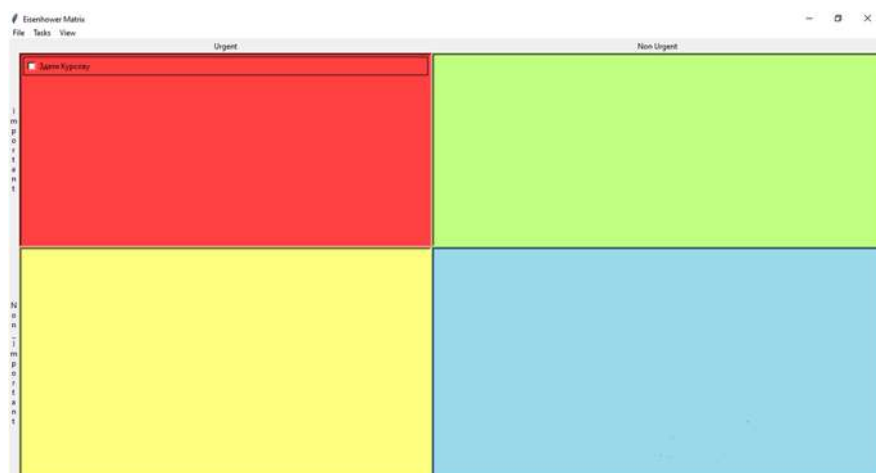
### **Додавання нових справ**

На рис. 4 показано додавання нової справи. Для нової справи потрібно вибрати ступінь важливості та терміновості.



*Рис. 4. Додавання нової справи*

На рис. 5 можна бачити, що справа додалась згідно з параметрами які їй надавались.



*Рис. 5. Справа додалась*

## Виконання справи

На рис. 6 відмічені ті справи, які ми вже виконали.

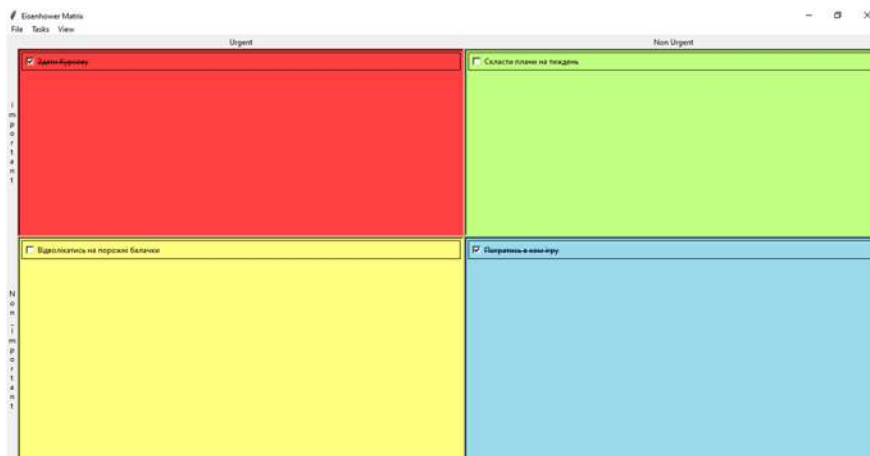


Рис. 6. Відмічені справи

## Приховання або видалення справ

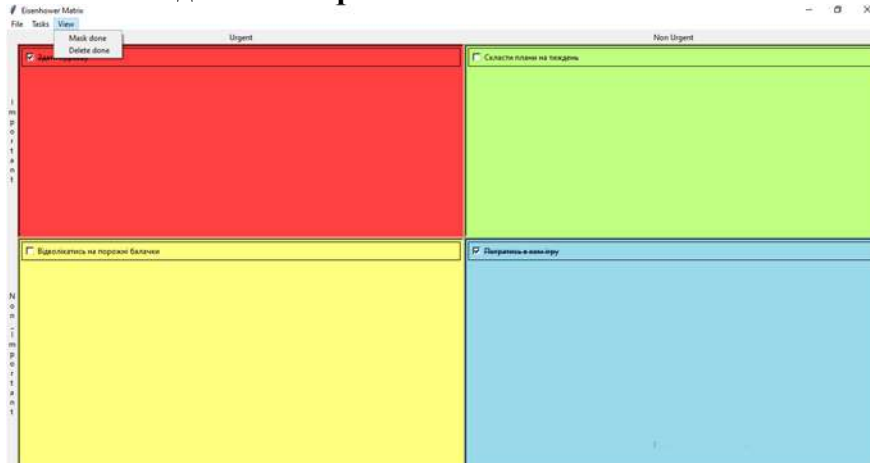


Рис. 7. Приховання або видалення справ

На рисунку 7 виділяються елементи, які потрібно або приховати, або видалити. Mask done – приховати елементи. Delete done – видалити елементи.

**Висновки.** В ході виконання роботи була докладно вивчена мова програмування «Python» та її бібліотека Tkinter. Вивчена робота бази даних Microsoft Access. Це дозволило створити десктоп-додаток для запису власних справ.

Для досягнення мети дослідження було вивчено методику проектування десктоп-додатків на мові програмування Python.

Постановка задачі полягала у розробці десктоп-додатку для планування власних справ. Для вирішення поставленої задачі було вивчено середовище розробки JetBrains PyCharm.

Логіка додатку побудована на основі матриці Ейзенхауера. Для цього всі справи поділяються на чотири категорії за ступенем важливості та терміновості. Для зручності кожна з цих категорій має свій колір. Також є можливість встановлення дати виконання кожної справи. Якщо справа виконана – її можна викреслити.

Був проведений порівняльний аналіз схожих десктоп-додатків. Оскільки їх виявилось мало, тому також був проведений огляд мобільних додатків.

*Анотація.* Стаття посвящена розробці програмного продукту для облегчення планирования собственных дел по степени важности и срочности. В работе приведены преимущества и недостатки данного

программного продукту. Практическая часть разработки – рабочий стол – приложение, разработанный для платформы Windows.

*Ключевые слова:* Python, десктоп-приложение, программный продукт, Матрица Эйзенхауэра.

*Abstract.* The article is devoted to the development of a software product to facilitate planning your own affairs in terms of importance and urgency. The paper describes the advantages and disadvantages of this software product. The practical part of development is the desktop – an application developed for the Windows platform.

*Key words:* Python, desktop application, software product, Eisenhower Matrix.

## СПИСОК ЛІТЕРАТУРИ

1. Swaroop C.H. A byte of Python
2. Функціональність PyCharm. URL: <https://www.jetbrains.com/ru-ru/pycharm/features/>
3. Що таке Tkinter. 2018. URL: <https://younglinux.info/tkinter/tkinter.php>
4. Навчання Python Gui. 2019. URL: <https://pythonru.com/uroki/obuchenie-python-gui-uroki-po-tkinter>
5. Уроки для вивчення Tkinter. 2019. URL: <https://python-scripts.com/tkinter-introduction>
6. Введення в Tkinter. 2011. URL: <https://habr.com/ru/post/133337/>
7. Реалізація MVC паттерна. 2018. URL: <https://habr.com/ru/post/150267/>
8. Що таке MVC: початкові концепції. 2019. URL: [https://skillbox.ru/media/code/chto\\_takoe\\_mvc\\_bazovye\\_kontseptsii\\_i\\_primer\\_prilozheniya/](https://skillbox.ru/media/code/chto_takoe_mvc_bazovye_kontseptsii_i_primer_prilozheniya/)
9. Як користуватись Microsoft Access. 2013. URL: <https://ru.wikihow.com/%D0%BF%D0%BE%D0%BB%D1%8C%D0%B7%D0%BE%D0%B2%D0%B0%D1%82%D1%8C%D1%81%D1%8F-Microsoft-Access>
10. Microsoft Access. 2010. URL:
11. URL: <https://sites.google.com/site/karnasiy1course/home/microsoft-access>

УДК 621.391.63/316.5

## РОЗРОБКА СЕРВІСУ РОЗПІЗНАННЯ МОВЛЕННЯ VAMBUSA

*І. О. Ставицький, А. Р. Нескородєва*

*Анотація.* Співробітниками міжкафедральної лабораторії з машинного навчання і інтелектуального аналізу даних був розроблений тестовий варіант сервісу Vambusa. Згаданий сервіс дозволить користувачу без втрати якості сприйняття обробляти значні обсяги аудіо та відео інформації. Сервіс Vambusa призначений для транскрибування мовлення у текст та генерації стислого переказу отриманого тексту (summary). Описані особливості архітектури мікросервісів застосунку, проведено порівняння із існуючими аналогами.

*Ключові слова:* обробка природної мови, розпізнавання мовлення, мікросервіс.

Обробка природної мови (NLP – Natural-language processing) є одним з найскладніших і разом з тим найперспективніших напрямів сучасного розвитку задач машинного навчання та аналізу даних (популярніше кажучи – задач штучного інтелекту, AI – Artificial intelligence). Згідно із сучасними визначеннями [1], обробку природної мови можна визначити як програмні та комп'ютерні додатки, створені для застосування обчислювальних методів для аналізу та синтезу природної мови та мовлення. Головна мета таких додатків полягає у реалізації обробки мови, подібно тому, як це робить людина, для аналізу в широкому сенсі текстів, сформованих за допомогою комп'ютерної техніки.

На сучасному етапі розвитку технологій NLP, серед головних завдань обробки природної мови можна виділити:

- *видобування даних* (data mining);
- *генеравання природної мови* (NLG – natural-language generation);
- *інформаційний пошук* (information retrieval);
- *машинний переклад* (machine translation);
- *розпізнавання мовлення* (speech recognition), або як технічний термін *мовлення-у-текст* (STT – speech to text);
- *синтез мовлення*;
- тощо.