User – абстрактний клас, який використовуються для перевірки коректності роботи програми та кращого її контролю. Методи, які визначені в ньому допомагають краще організувати роботу програми, уникнути певної кількості помилок, або легше діагностувати їх при не коректній роботі програми. Серед методів присутній метод \_\_init\_\_ який буде виконувати ініціалізацію об’єктів класів, які будуть наслідувати User, так як сам клас є абстрактним, також присутній один метод, що допомагає з перевіркою коректності роботи програми та чотири методи що покращують її використання та спрощують роботу з частинами програми, написаними на фреймворку Kivy.

check\_screen – метод, який визначає яка сторінка з графічного інтерфейсу програми на даний момент відображається користувачу. Цим самим він дозволяє визначитись з наступним алгоритмом дій. Метод повертає назву поточної сторінки.

Приклад використання:

>>>print(check\_screen())

>>>’Screen1’

change\_screen – метод, який виконує перехід між сторінками графічного інтерфейсу. Виконується після певних дій користувача ( наприклад натискання кнопки ) та відображає наступну сторінку, яка потрібна для продовження роботи в програмі. Метод приймає об’єкт класу який наслідує User та сторінку на яку має бути здійснений перехід. Використовує метод check\_screen для перевірки коректності своєї роботи. Метод повертає True у випадку якщо перехід відбувся вдало та False в іншому випадку.

Приклад використання:

>>>change\_screen(‘Screen2’)

>>>True

change\_login – метод, який змінює логін користувача, тобто gmail акаунт з якого буде відправлятись повідомлення з локацією користувача. Може бути використаний при виході з програми або зміні користувача. Методи приймає об’єкт класу який наслідує User та новий логін (gmail пошту). Після нього має бути викликаний інший метод – change\_pasword. Метод повертає True якщо логін успішно змінено та False в іншому випадку.

Приклад використання:

>>>change\_login(‘kursa4@gmail.com’)

>>>True

change\_password – метод, який змінює пароль користувача, тобто пароль від gmail акаунту користувача ( метод змінює не пароль на самому gmail а лише той який він сам буде використовувати при вході на пошту). Може бути використаним після зміни паролю від самого gmail акаунту користувача або після виконання методу change\_login. Метод приймає об’єкт класу який наслідує User та новий пароль та повертає True у випадку успішної зміни паролю або False у іншому випадку.

Приклад використання:

>>>change\_password(‘12345678’)

>>>False

change\_recipient – метод, який змінює отримувача локації користувача у випадку низького заряду телефона користувача, тобто змінює gmail пошту на яку буде відправлене повідомлення. Може бути використаним при переході отримувача на інший акаунт або втрати доступу до поточного, також при бажанні змінити отримувача чи взагалі забрати його. Метод приймає об’єкт класу який наслідує User та нову поштову адресу(на яку відтепер буде відправлятися повідомлення). Повертає True в разі успішної заміни отримувача та False у іншому випадку.

Приклад використання:

>>>change\_recipient(‘nokursa4@gmail.com’)

>>>True