УДК 004.4

ПРОЕКТИРОВАНИЕ И РАЗРАБОТКА ПРОГРАММНОГО ПРОДУКТА «ЛИЧНЫЙ БЛОКНОТ ДЛЯ ЗАПИСИ МЫСЛЕЙ»

Слепов В.А.

Нижнетагильский технологический институт (филиал) ФГАОУ ВО «УрФУ имени первого Президента России Б.Н. Ельцина», Нижний Тагил, e-mail: vlad.slepov@gmail.com

Актуализируется необходимость разработки личного блокнота для записи мыслей. Определяются этапы разработки предложенного программного продукта: изучение предпочтений пользователей, определение требований к программе и проектирование, разработка графического интерфейса программы, разработка приложения, оформление необходимой документации. Для анализа предпочтений пользователей предложен вариант набора вопросов социологического опроса для определения актуальности разработки. Полученные результаты опроса обработаны и продемонстрированы посредством диаграмм. Формирование общих требований к программному продукту показано с помощью диаграммы прецедентов. Диаграмма прецедентов определяет особенности программного продукта, отображает все способы взаимодействия пользователя: авторизация в приложении, создание, редактирование и удаление записей. Построена диаграмма активности, которая позволяет детализировать особенности реализации программным продуктом операций и прецедентов. Графический интерфейс создан средствами набора расширений графического фреймворка Qt для языка программирования Python. Выполнено тестирование программного продукта, результаты тестирования показали работоспособность продукта, все запланированные на этапе проектирования функции выполняются без ошибок. Предложен вариант развития проекта – создание мобильной версии. Описанный пример применения технологий проектирования и разработки продукта может быть использован при создании подобных программных решений.

Ключевые слова: программный продукт, проектирование, диаграмма прецедентов, диаграмма активности, разработка, Python, PyQt

DESIGN AND DEVELOPMENT OF THE PROGRAM PRODUCT «PERSONAL BLOCK FOR RECORDING THOUGHTS»

Slepov V.A.

Nizhny Tagil Technological Institute (branch) of Federal State Autonomous Educational Institution of Higher Education «Ural Federal University named after the first President of Russia B.N. Yeltsin», Nizhny Tagil, e-mail: vlad.slepov@gmail.com

The need to develop a personal notebook for recording thoughts is being updated. The stages of development of the proposed software product are determined: the study of user preferences, the definition of requirements for the program and design, the development of the graphical interface of the program, application development, design of the necessary documentation. To analyze user preferences, a variant of a set of questions of a sociological survey is proposed to determine the relevance of the development. The survey results were processed and demonstrated through diagrams. The formation of general requirements for a software product is shown using the use case diagram. The use case diagram defines the features of the software product, displays all the ways the user interacts: authorization in the application, creating, editing and deleting entries. An activity diagram is constructed that allows you to detail the features of the implementation of the software product operations and use cases. The graphical interface was created by means of the set of extensions of the Qt graphical framework for the Python programming language. The software product was tested, the test results showed the performance of the product, all the functions planned at the design stage are performed without errors. A project development option is proposed – the creation of a mobile version. The described example of application of product design and development technologies can be used to create similar software solutions.

Keywords: software product, design, use case diagram, activity diagram, development, Python, PyQt

В настоящее время пользователи используют компьютеры для решения самых разнообразных задач: планирования дел, организации досуга, передачи нужной информации и др. Компьютер – это незаменимый помощник человека в быстром и эффективном выполнении его функций. При этом в процессе работы в Интернете часто пользователям необходимо сделать какие-то заметки, сохранить интересные цитаты, данные для доступа на тот или иной сайт и др. Существует ряд работ, посвященных проблеме определения функциональности программ, позволяющих вести записи, учет и планирование времени, выбора соответствующих технологий разработки. Автор [1] предлагает программу «Weekly planner», которая позволяет выполнять привязанные к календарю записи и предназначена для выполнения на смартфонах или планшетах. В работе [2] предлагается система «On-line ежедневник», доступ к которой возможен для нескольких пользователей с различными правами доступа: от простого просмотра до изменения и удаления записей. Особенностью предлагаемой

в данной статье программы является возможность защищенного хранения важной информации. Кроме того, данное приложение может быть использовано в качестве ежедневника с функцией напоминания об определенном событии.

Цель исследования заключается в описании процессов проектирования, разработки и тестирования приложения для ведения заметок «Личный блокнот для записи мыслей». Программа должна быть разработана на языке программирования Python с использованием фреймворка Qt [3]. Отметим, что реализация действий на этапах проектирования, разработки и тестирования является универсальной и может быть интересна разработчикам любых программных продуктов.

Материалы и методы исследования

Решение данной задачи осуществлялось в несколько этапов, которые соответствуют требованиям ГОСТ [4; 5].

1. Изучить предпочтения пользователей.
2. Выполнить проектирование и определить требования к программе.
3. Разработать графический интерфейс для программы.
4. Разработать приложение.
5. Протестировать программу.
6. Оформить необходимую программную документацию.

Рассмотрим каждый из перечисленных выше этапов более подробно. На первом этапе был проведен опрос респондентов с целью выяснения их мнения о разрабатываемом программном продукте. Для того чтобы опросить большее количество респондентов, была разработана анкета в Google Forms. Данная анкета была предложена пользователям разного возраста. Результаты анкетирования использовались для выделения основных функций приложения.

На этапе проектирования были визуализированы действия пользователя при работе с программой. Визуализация осуществлялась на основе языка UML, который используется для графического описания процессов. Кроме того, на данном этапе были определены требования к программе и, в соответствии с данными требованиями, выполнен отбор средств разработки.

Процесс разработки программы условно был разделен на два этапа: на первом этапе создавался интерфейс, а на втором – написан программный код. Для разработки программы использовался язык Python. В PyQt создавался интерфейс, затем он перекомпилировался в формат .py. Вся дальнейшая разработка осуществлялась на языке Python в интегрированной среде IDLE.

Далее программа была протестирована, результаты тестирования оформлены в виде тест-кейсов. Все ошибки, выявленные в процессе тестирования, были исправлены. На заключительном этапе был оформлен документ «Руководство пользователя».

Результаты исследования и их обсуждение

Рассмотрим процедуру разработки приложения. На первом этапе был проведен социологический опрос для оценки актуальности проекта. Пользователям разного возраста были заданы два вопроса.

1. Пользуетесь ли Вы приложениями для ведения заметок?
2. Какие критерии важны для Вас при выборе приложения для ведения заметок?

Результаты распределения ответов на опрос представлены на рис. 1–2.

Среди всех анкетируемых 45 % отметили, что они постоянно ведут ежедневник и/или составляют список дел. На основе полученных данных можно сделать вывод, что такое приложение будет пользоваться спросом у пользователей, тем более если оно будет установлено на компьютер или ноутбук, за которым человек проводит много времени.

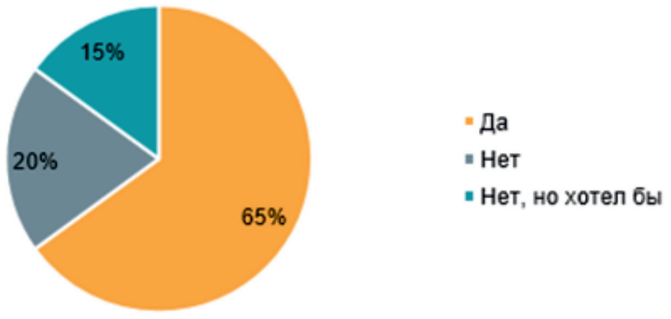

Рис. 1. Результаты ответа на вопрос «Пользуетесь ли Вы приложениями для ведения заметок?»

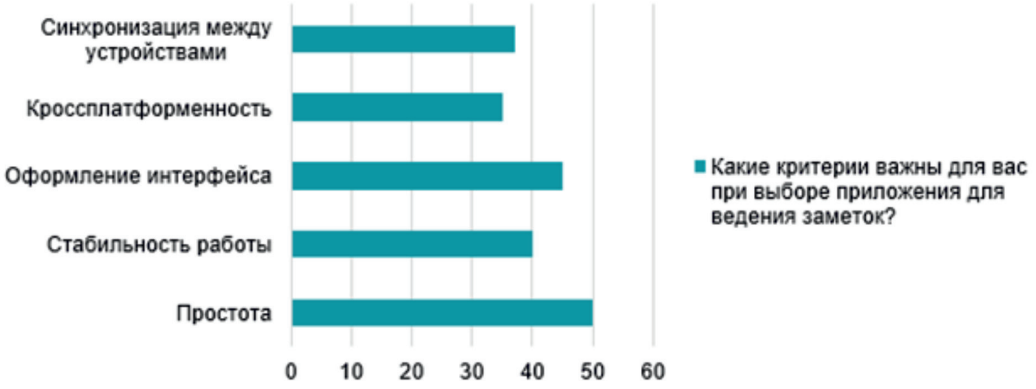

Рис. 2. Результаты ответа на вопрос «Какие критерии для Вас важны при выборе приложения для ведения заметок?»

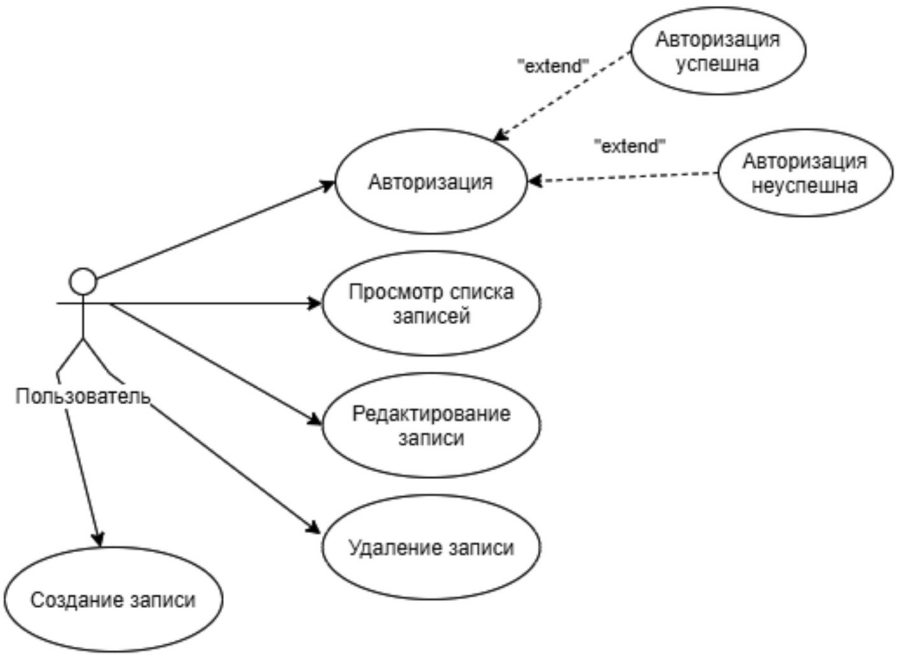

Рис. 3. Диаграмма прецедентов

Далее было выполнено проектирование для выделения основных функций будущего программного продукта, формирования требований, разработки концептуальной модели системы для ее последующей детализации [6–8], которые отображены с помощью диаграммы прецедентов (рис. 3). Диаграмма прецедентов может быть использована для определения способов использования программных продуктов, предназначенных для автоматизации задач из самых разнообразных предметных областей [9–11].

Согласно диаграмме прецедентов пользователь имеет возможность зайти в приложение со своим логином и паролем, создать нужную запись. Также он может редактировать, удалять и добавлять заметки.

Далее была создана диаграмма активности, которая позволяет детализировать особенности реализации программным продуктом операций и прецедентов. На рис. 4 показана детализация прецедента «Авторизация»: после запуска приложения пользователь видит окно для ввода ло-

гина и пароля, вводит соответствующую информацию. Если авторизация прошла успешно, пользователь может добавлять новые заметки, редактировать и удалять уже существующие.

Этап проектирования позволил сформулировать набор функций программы:

– оформление заметок в различные дни, с привязкой к календарю;

– использование как ежедневник для напоминания о важных событиях;

– вывод информации о событиях;

– защита данных паролем.

Интерфейс программы был выполнен с помощью PyQt5 [12; 13]. Это набор библиотек для Python для создания графического интерфейса. Для его установки в командной строке необходимо набрать команду *pip3 install pyqt5*, далее запустить Designer и создать графический интерфейс пользователя. Графический интерфейс должен быть максимально удобным и понятным для пользователя. Отметим, что создание графического интерфейса посредством Designer предполагает работу с такими элементами, как диалоговые окна, кнопки, метки и поля. В дальнейшем, уже непосредственно в среде программирования для Python, к данному интерфейсу необходимо добавить функции обработки данных. Структура программного продукта представлена на рис. 5.

В папке dist находится исполняемый файл, чтобы не требовалось установки Python и библиотек для запуска приложения. Для компиляции в EXE файл использовалась библиотека Pyinstaller. В папке script расположены файлы интерфейса, импортированные в Python из Qt Designer с помощью pyuic5. В папке UI находятся исходные XML-файлы описания интерфейса, созданные в Qt Designer. В Diary.py прописан исходный код программы.

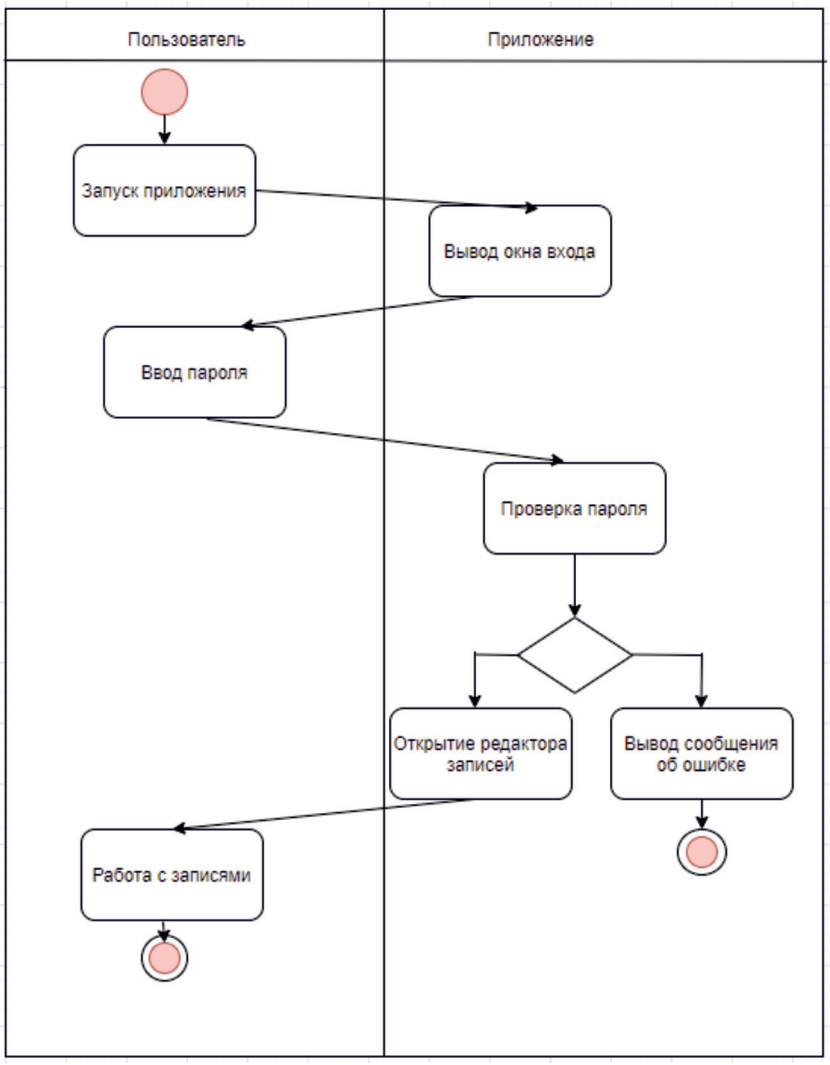

Рис. 4. Диаграмма активности

Для кнопки в главном окне, в котором хранится список всех заметок, задана функция-обработчик, которая вызывает окно создания заметки, а дальше управление передаётся согласно его внутренней логике. При двойном клике на любой из существующих заметок срабатывает обработчик, который вызывает окно редактирования.

Интерфейс программы показан на рис. 6.

Рис. 5. Структура программы

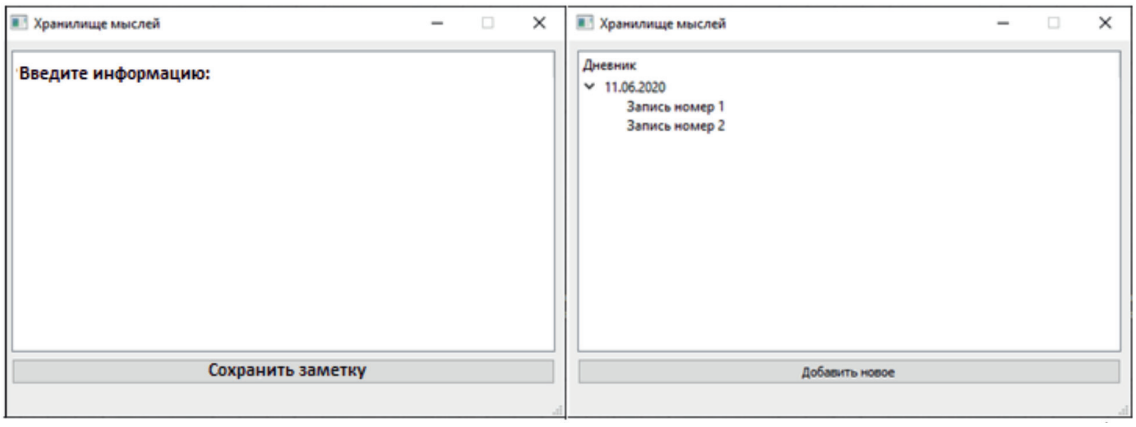

Рис. 6. Пример реализации в Python

Результаты тестирования программы

Вид теста	Содержание теста	Результат теста
Регистрация в программе	1. Нажать на кнопку «Регистрация». 2. Ввести необходимые параметры. 3. Нажать на кнопку «Сохранить». 4. Проверить вход в программу (см. п. «Организация входных данных»)	Выполнен
Организация входных данных	1. Ввести логин, который состоит из английских букв и цифр. 2. Ввести пароль, который включает только цифры. 3. Выполнить вход в программу. 4. Если программа выдает ошибку, вернуться к пункту 1	Выполнен
Открытие окна для создания заметок	1. Нажать на кнопку «Добавить новое». 2. Ввести информацию. 3. Нажать кнопку «Сохранить заметку»	Выполнен
Редактирование существующей заметки	1. Открыть существующую заметку. 2. Выбрать «Карандаш». 3. Внести изменения. 4. Сохранить изменения	Выполнен
Удаление заметки	1. Нажать на кнопку «Удалить заметку». 2. Подтвердить удаление заметки. 3. Нажать на кнопку «ОК». 4. Вернуться к полю для создания новой заметки	Выполнен

На последнем этапе было проведено тестирование программного продукта методом черного ящика. Метод черного ящика заключается в проверке функционального поведения программы как результата всех возможных действий пользователя. Итоги тестирования представлены в таблице.

«Личный блокнот для записей мыслей» разработан с учетом пожеланий будущих пользователей. Он имеет предпосылки для дальнейшего развития, а именно планируется создание мобильной версии приложения. Кроме того, планируется доработать еще одну важнейшую функцию данного программного продукта – возможность сохранять в заметках фотографии.

Заключение

Описанный пример применения технологий проектирования и разработки продукта может быть использован при создании подобных программных продуктов. Кроме того, этапы разработки данной программы могут быть интересны для всех пользователей, разрабатывающих программные продукты на языке Python.

Список литературы

1. Димитриев Р.С. Weekly planner. Свидетельство о регистрации программы для ЭВМ RU 2016614019, 12.04.2016. Заявка № 2015662492 от 18.12.2015.

2. Алексеев С.А. On-line ежедневник – электронная система планирования времени // Молодые ученые – промышленности, науке, технологиям и профессиональному образованию: проблемы и новые решения: сборник научных докладов VII Международной научно-практической конференции. 2007. С. 300–303.

3. Бэрри Пол. Изучаем программирование на Python. М.: Эксмо, 2017. 624 с.

4. ГОСТ 19.201-78. Техническое задание. Требования к содержанию и оформлению. [Электронный ресурс]. URL: http://docs.cntd.ru/document/gost-19-201-78 (дата обращения: 12.06.2020).

5. ГОСТ Р ИСО/МЭК ТО 15271-2002. Информационная технология. Руководство по применению ГОСТ Р ИСО/МЭК 12207 (Процессы жизненного цикла программных средств). [Электронный ресурс]. URL: http://docs.cntd.ru/document/gost-19-201-78 (дата обращения: 12.06.2020).

6. Васева Е.С., Смирнов М.С. Виртуальная экскурсия по образовательному учреждению: особенности проектирования // Наука и перспективы. 2018. № 1. С. 68-74.

7. Ведерников Н.А., Бужинская Н.В. Применение языка программирования С++ для разработки электронных образовательных ресурсов по математике // Научное обозрение. Технические науки. 2019. № 3. С. 15–20.

8. Смагина Н.В. Автоматизация деятельности кондитерского производства с использованием 1С: Предприятия // Ростовский научный журнал. 2016. № 6. С. 41–53.

9. Драчев В.В., Бужинская Н.В., Васева Е.С. Разработка программного решения для автоматизации оперативного изменения контента сайтов, созданных с помощью CMS WordPress // Программная инженерия. 2019. Т. 10. № 7–8. С. 334–343.

10. Широбокова С.Н., Бабеев М.С. Визуальное UML-моделирование деловых процессов управления бухгалтерского учета вуза // Теория, методы проектирования, программно-техническая платформа корпоративных информационных систем: материалы 14-ой Международной научно-практической конференции, посвященной 110-летию Южно-Российского государственного политехнического университета (НПИ) имени М.И. Платова. 2016. С. 148–154.

11. Шурыгин И.С., Филимонов В.Ю., Вдовенко В.В. Диаграмма прецедентов информационной системы исследования и прогнозирования состояния геологических резервов // Актуальные проблемы авиации и космонавтики. 2011. Т. 1. № 7. С. 415–416.

12. Горожанов А.И. PyQt 5 для лингвистов: профессионально-ориентированное программирование. Электронное учебное пособие. [Электронный ресурс]. URL: https://pyqtforlinguists.appspot.com/book.pdf (дата обращения: 12.06.2020).

13. Жуков Р.А. Язык программирования Python: практикум. Учебное пособие для студентов высших учебных заведений и обучающихся по направлению подготовки 38.03.05 «Бизнес-информатика» (квалификация (степень) «бакалавр»). М.: ИНФРА-М, 2019. 214 с.