

- Zboriv, Ukraine, October 02-04, 2024. CEUR Workshop Proceedings, 2024, 3842, pp. 147-156.
2. Raspberry Pi Pico W. Arduino.ua. <https://shorturl.at/CoKac> (дата звернення 10.05.2025)
 3. RPP-316 PD20W+QC22.5W. <https://shorturl.at/aQxYF> (дата звернення 11.05.2025)
 4. Brown M., 2011. Power Sources and Supplies: World Class Designs. Newnes, 368 с.
 5. Акумулятори 18650. <https://shorturl.at/cmNGb> (дата звернення 13.05.2025)

УДК 004.41

С.І. Подедвірний, Г.Б. Цуприк, к.т.н., доц.

Тернопільський національний технічний університет імені Івана Пулюя, Україна

ІНТЕГРАЦІЯ ДЕСКТОПНОГО РОЗКЛАДУ З GOOGLE CALENDAR ЗА ДОПОМОГОЮ C++/QT ТА GOOGLE API

S.I. Podedvirniy, H.B. Tsupryk, Ph.D., Assoc. Prof.

INTEGRATION OF DESKTOP WALLOW WITH GOOGLE CALENDAR USING C++/QT AND GOOGLE API

У сучасному цифровому середовищі персональне планування та управління часом набувають все більшого значення як у професійному, так і в особистому контексті. Хоча існує велика кількість вебсервісів для планування подій, багато користувачів виявляють потребу у функціональному десктопному застосунку, який би поєднував автономну роботу з можливістю хмарної синхронізації. Саме така концепція стала основою для створення програмного засобу — десктопного планувальника розкладу, розробленого із використанням кросплатформного фреймворку Qt (мова програмування C++) та інтегрованого з Google Calendar API.

Метою роботи стало створення зручного інтерфейсу для управління подіями, який дозволяє створювати, редагувати та видаляти події локально, а також здійснювати двосторонню синхронізацію з Google Calendar. Це дозволяє користувачеві мати доступ до своїх подій з будь-якого пристрою, водночас маючи стабільний інструмент для роботи в офлайн-режимі. Реалізація проєкту базується на поєднанні кількох ключових технологічних компонентів: Qt як інструменту для створення графічного інтерфейсу користувача, бібліотек для мережевої взаємодії (зокрема QNetworkAccessManager), та Google Calendar API, що дозволяє здійснювати запити до сервера за протоколом REST.

Особливу увагу було приділено реалізації процедури аутентифікації за допомогою протоколу OAuth 2.0. Цей етап передбачає відкриття сторінки авторизації в браузері, отримання коду доступу та подальше використання маркера доступу (access token) для виконання запитів до API. Таким чином, забезпечується безпечний обмін даними без потреби зберігати облікові дані користувача в системі. Комунікація із сервером Google Calendar здійснюється за допомогою HTTP-запитів (GET, POST, PATCH, DELETE), а результати обробляються у форматі JSON із використанням вбудованих можливостей Qt для роботи з цим форматом (класи QJsonDocument, QJsonObject тощо).

Архітектура застосунку реалізована за принципами Model-View-Controller (MVC), де логіка обробки подій відокремлена від відображення інтерфейсу. Це забезпечує зручність масштабування та супроводу коду. Дані календаря зберігаються у вигляді локального кешу з використанням формату JSON, що дозволяє забезпечити роботу в офлайн-режимі та синхронізувати зміни при відновленні підключення до Інтернету.

Інтерфейс користувача спроектований відповідно до принципів простоти та ефективності. Основна панель дозволяє переглядати події за днем або тижнем, забезпечуючи зручну навігацію за допомогою календаря та візуалізації у вигляді таймлайнів або таблиці. Для кожної події можна задавати назву, опис, час початку та завершення, а також вибирати категорію, яка відображається різними кольорами. Передбачено інтерактивну навігацію,

гарячі клавіші, сповіщення про майбутні події, а також підтримку кількох облікових записів Google одночасно.

У процесі реалізації виникло кілька викликів, серед яких — правильна обробка токенів доступу та їх оновлення, обмеження на частоту запитів до API (rate limits), а також забезпечення стабільної роботи при нестабільному з'єднанні. Усі ці аспекти були враховані шляхом реалізації механізмів повторної спроби (retry), логування помилок і зручного інформування користувача про стан синхронізації.

Запропоноване рішення має переваги перед традиційними веб календарями, адже забезпечує повну функціональність навіть без доступу до Інтернету, має швидкодіюче нативне виконання, а також гнучкий інтерфейс, який можна адаптувати під конкретні потреби користувача. У порівнянні з існуючими аналогами, такими як Google Calendar Web, Microsoft Outlook чи Thunderbird Calendar, даний застосунок поєднує простоту десктопного інтерфейсу з перевагами хмарної інтеграції.

У перспективі розробка може бути доповнена синхронізацією з іншими сервісами календарів, а також інтеграцією з локальними базами даних (наприклад, SQLite) для зберігання історії змін та підтримки командної роботи. Можливе також впровадження штучного інтелекту для автоматичного розподілу подій або рекомендацій щодо оптимізації розкладу.

Таким чином, у роботі було створено повноцінний десктопний інструмент планування часу, який поєднує зручність локального використання з перевагами хмарної синхронізації. Результати свідчать про ефективність інтеграції C++/Qt із сучасними веб сервісами на прикладі Google Calendar API, що відкриває широкі перспективи для створення універсальних засобів персонального тайм-менеджменту, які відповідають вимогам сучасного користувача.

Література

1. Google Calendar API Overview | Calendar API | Google Developers. Google Developers. URL: <https://developers.google.com/calendar> (дата звернення: 19.05.2025).
2. RFC 6749: The OAuth 2.0 Authorization Framework / D. Hardt. IETF, 2012. URL: <https://datatracker.ietf.org/doc/html/rfc6749> (дата звернення: 19.05.2025).
3. Qt Documentation | Application Development with Qt | Qt Company. URL: <https://doc.qt.io/qt-6/> (дата звернення: 19.05.2025).
4. Methods of constructing algorithms for comparative test statistical verification of mathematical models of bioobject responses to low-intensity stimuli / Bohdan Yavorsky, Evhenia Yavorska, Halyna Tsupryk, Roman Kinash // Scientific Journal of TNTU. — Tern.: TNTU, 2023. — Vol 112. — No 4. — P. 82–90.
5. Бойко І.В., М.Р. Петрик, Г.Б. Цуприк. Інформаційні технології видобутку даних (Data mining, високопродуктивні обчислення у складних системах): навчальний посібник. Тернопіль: : ТНТУ 2020 – 62 с.
6. Яворська, Є.Б., Кінаш, Р.В., Цуприк, Г.Б., Николайчук В.І. Проблеми виявлення біосигналів у медичних інформаційних системах / Є.Б. Яворська, Кінаш Р.В., Г.Б. Цуприк, В.І. Николайчук // IV Міжнародна науково-практична конференція «Інформаційні системи та технології в медицині» (ICM–2021) [Текст] : зб. наук. пр. – Харків : Нац. аерокосм. ун-т ім. М. Є. Жуковського «Харків. авіац. ін-т», 2021. – 25-26 с.