

Qownnotes

Christian Külker

2024-06-20

Contents

1	Introduction	2
2	Building on Debian 12	2
2.1	Dependencies	2
2.2	Source	2
2.3	Build	2
2.4	Install	3
3	Ansible Source Build Install	3
4	Debug	6
4.1	General Information	6
4.2	Spellchecking	7
4.3	Note folders	7
4.4	Note folder default	7
5	Links	7
6	Videos	8
7	Critique	8
7.1	Pros	8
7.2	Cons	8
8	History	8
9	Disclaimer of Warranty	9
10	Limitation of Liability	9

1 Introduction

QOwnNotes is a note-taking application designed with a focus on user independence and efficiency. It stands out by saving notes as plain-text markdown files on the user's own computer, thereby avoiding the potential issues of vendor lock-in seen in some other platforms. The application is designed for quick performance and is conscious of system resource usage, striving to keep its processor and memory usage low. QOwnNotes offers customization and scripting options, allowing users to adjust it to their workflow preferences. It includes additional functionalities such as real-time markdown preview, integrated spellchecking, support for tabbed editing, and a Vim mode, which enhance its practicality.

2 Building on Debian 12

2.1 Dependencies

```
aptitude install qtchooser libqt5websockets5 libqt5websockets5-dev \  
libqt5x11extras5 libqt5x11extras5-dev  
# OR (not tested)  
# aptitude install qmake libqt6websockets6 libqt6websockets6-dev \  
# libqt6x11extras6 libqt6x11extras6-dev  
↪ qml6-module-qt5compat-graphicaleffects
```

2.2 Source

```
git clone https://github.com/pbek/QOwnNotes.git -b release  
cd QOwnNotes  
git submodule update --init  
cd src
```

2.3 Build

```
# Build binary translation files if you want another language than English  
# (not tested)  
# lrelease QOwnNotes.pro  
  
# Prepare build process  
qmake  
Project MESSAGE: Currently in RELEASE mode.
```

```
# Build the application
make
[...]
```

2.4 Install

This will install **not** into `/usr/local` but directly into `/usr` if no `PREFIX` variable is set via `qmake` and it might mess with packages! Directories affected are: `/usr/{bin,share}`. It is also possible to run the application directly from the `src` directory without installing it. (`./QOwnNotes`)

```
cd src
make install
```

3 Ansible Source Build Install

This Ansible cookbook will

- Install dependencies
- Create some directory structure
- Clone the repository
- Initialize the sub-modules inside the repository
- Configure QOwnNotes to be installed in `/srv/qowennotes`
- Build
- Install
- Make it accessible under `zsh`

It uses the function `path_add_after` to add the application to the current path, this function should be available in your ZSH configuration `zshrc` globally. Here it is for adding a new path before and after.

```
path_add_before(){
    if [ -d "$1" ] && [[ ":$PATH:" != *"$1:"* ]]; then
        path=($1 $path)
    fi
}
path_add_after(){
    if [ -d "$1" ] && [[ ":$PATH:" != *"$1:"* ]]; then
        path+=($1)
    fi
}
```

```
fi
}
```

Then the application PATH is set for all users, made available for Ansible inside the variable `users` like this:

```
users:
- name: $USER
  group: $GROUP
  home: /home/$USER
```

Replace `$USER` and `$GROUP` with real values and add more entries like this to the list in case you have more than one user.

```
- name: Install and setup QOwnNotes
  hosts: all
  become: yes # This is required for installing packages and writing to
  /srv
  gather_facts: yes # Because of {{ansible_processor_vcpus}}
  vars:
    ns: qownnotes
    cores: "{{ansible_processor_vcpus}}"
    bin: QOwnNotes
    dst: "/srv/{{ns}}"
    src: "{{github.dir}}/pbek/{{bin}}/src"
    my_user: "{{user.user}}"
    my_group: "{{user.group}}"
  packages:
    - qtchooser
    - libqt5websockets5
    - libqt5websockets5-dev
    - libqt5x11extras5-dev
    - libqt5x11extras5
  tasks:
    - name: "{{ns}}: Install and update packages"
      package:
        name: "{{ packages }}"
        state: latest
    - name: "{{ns}}: Create directory for {{bin}} source"
      file:
        path: "{{github.dir}}/pbek"
        recurse: true
        owner: "{{my_user}}"
```

```
    group: "{{my_group}}"
    state: directory
    mode: '0755'
- name: "{{ns}}: Clone '{{bin}}' repository to
↳ '{{github.dir}}/pbek/{{bin}}'"
  git:
    repo: 'https://github.com/pbek/{{bin}}.git'
    dest: '{{github.dir}}/pbek/{{bin}}'
    version: release
    force: yes # In case build was not successful
    clone: yes
    update: yes
  become: yes
  become_user: "{{my_user}}"
  when: not (github.dir + '/pbek/' + bin + '/.git') is directory
- name: "{{ns}}: Initialize git submodules
↳ '{{github.dir}}/pbek/{{bin}}'"
  shell: git submodule update --init
  args:
    chdir: "{{github.dir}}/pbek/{{bin}}"
  become: yes
  become_user: "{{my_user}}"
  when: not (github.dir + '/pbek/' + bin + '/.git/modules') is
↳ directory
- name: "{{ns}}: Configure {{bin}} prefix '{{dst}}'"
  shell: "qmake PREFIX={{dst}}"
  args:
    chdir: "{{src}}"
  become: yes
  become_user: "{{my_user}}"
  when: not (dst + '/bin/' + bin) is file
- name: "{{ns}}: Build {{bin}} with {{core}} cores at '{{src}}'"
  shell: "make -j {{cores}}"
  args:
    chdir: "{{src}}"
  become: yes
  become_user: "{{my_user}}"
  when: not (dst + '/bin/' + bin) is file
- name: "{{ns}}: Install {{bin}} to '{{dst}}'"
  shell: make install
  args:
    chdir: "{{src}}"
```

```
become: yes
become_user: "{{my_user}}"
when: not (dst + '/bin/' + bin) is file
- name: "{{ns}}: Add dir '{{dst}}/bin' to PATH for each user"
  copy:
    content: "path_add_after {{dst}}/bin"
    dest: "{{item.home}}/.zsh/829_{{ns}}.zsh"
    owner: "{{item.name}}"
    group: "{{item.group}}"
    mode: 0644
  loop: "{{users}}"
become: yes
```

4 Debug

The application provides a debug tab inside the application settings. Here some, but not all information.

4.1 General Information

```
**Current Date**: `Thu Feb 8 21:18:02 2024`
**Version**: `24.2.0`
**Build date**: `Feb 8 2024`
**Build number**: `1077`
**Platform**: `linux`
**Operating System**: `Debian GNU/Linux 12 (bookworm)`
**Build architecture**: `x86_64`
**Current architecture**: `x86_64`
**Release**: `Self-build`
**Qt Version (build)**: `5.15.8`
**Qt Version (runtime)**: `5.15.8`
**Portable mode**: `no`
**Settings path / key**: `~/.config/PBE/QOwnNotes.conf`
**Application database path**:
  ↳ `~/.local/share/PBE/QOwnNotes/QOwnNotes.sqlite`
**Application arguments**: `./QOwnNotes`
**Qt Debug**: `0`
**Locale (system)**: `en_US`
**Locale (interface)**: `empty`
**Primary screen resolution**: `1920x1200`
```

```
**Screen resolution(s)**: `1920x1200, 1200x1920, 1920x1200`  
**Icon theme**: `mate`  
**Notes in current note folder**: `2`  
**Calendar items**: `0`  
**Enabled scripts**: `0`
```

4.2 Spellchecking

```
**Enabled**: `true`  
**Selected language**: `auto`  
**Language codes**: `en_US`  
**Language names**: `American English (United States)`  
**Application dictionaries path**:  
↳ `/$HOME/.local/share/PBE/QOwnNotes/dicts`
```

4.3 Note folders

The note folder will be chosen at application startup.

```
**currentNoteFolderId**: `1`
```

4.4 Note folder default

```
**id**: `1`  
**isCurrent**: `yes`  
**activeTagId**: `0`  
**localPath**: `$HOME/Notes` # user choose this value  
**remotePath**: `Notes`  
**cloudConnectionId**: `1`  
**isShowSubfolders**: `yes`  
**isUseGit**: `no`  
**allowDifferentNoteFileName**: `no`  
**activeNoteSubFolder name**: *empty*  
**database file**: `$HOME/Notes/notes.sqlite`
```

5 Links

- Home: <https://www.qownnotes.org/>
- Source: <https://github.com/pbek/QOwnNotes>

6 Videos

- A video about choosing a note taking application <https://www.youtube.com/watch?v=XRpHla-2XCE>

7 Critique

QOwnNotes is a versatile note-taking application designed for those who prefer working with Markdown files. It offers a blend of simplicity and functionality, catering to users who need a straightforward yet capable note-taking tool.

7.1 Pros

- Uses plain Markdown files
- The markdown preview is interactive: Checking a TODO point also marks it as 'done' in the source for example
- Easy compilation
- Git support (not tested)
- FOSS
- Encryption (not tested)
- Tags
- Web application support (not tested)
- Can use a generic CalDAV server (not tested)

7.2 Cons

- Install to `/usr` instead of `/usr/local` without PREFIX
- Uses many SQL lite databases (for meta data)
- Configuration data at several locations (`$HOME/.local`, `$HOME/.config`, `$HOME-Notes`)
- Some features (like versioning) only works with cloud service
- Binary is spelled mixed case including upper case

8 History

Version	Date	Notes
0.1.1	2024-06-20	Add debug section and Ansible playbook
0.1.0	2023-09-14	Initial release

9 Disclaimer of Warranty

THERE IS NO WARRANTY FOR THIS INFORMATION, DOCUMENTS AND PROGRAMS, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE INFORMATION, DOCUMENT OR THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE INFORMATION, DOCUMENTS AND PROGRAMS IS WITH YOU. SHOULD THE INFORMATION, DOCUMENTS OR PROGRAMS PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

10 Limitation of Liability

IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MODIFIES AND/OR CONVEYS THE INFORMATION, DOCUMENTS OR PROGRAMS AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE INFORMATION, DOCUMENTS OR PROGRAMS (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE INFORMATION, DOCUMENTS OR PROGRAMS TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.