

WowApp Coding Challenge

Please, provide production level solution.

The task requires knowledge of Qt 5 (Qt Widgets, Qt Core, Qt Network, Qt Concurrent).

The final solution will be judged on:

- coding style;
- code architecture;
- overall Qt usage;
- code efficiency;
- proper usage of C++;
- code readability.

We don't set any timing boundaries regarding task completion, but please, provide estimate for the task completion. It's needed to see how your estimates are reliable.

With that being said, have fun and good luck!

The task

Your goal is to create an application which will download a list of contacts (roster) and present it using Qt Widgets.

The data

The roster has to be downloaded from <https://file.wowapp.me/owncloud/index.php/s/sGOXibS0ZSspQE8/download> if it's not already been cached locally.

The roster is represented as a JSON object which had the following format:

- roster** — an array of items;
- rosterSize** — the size of the roster.

The array's elements have the following format:

- id** — unique id of the roster's item;
- group** — the name of the group give items blogs to;
- groupOrder** — the order of the group in the resulting list;
- account** — contains detailed information about the item. It has following data:
 - userName** — user name;
 - firstName** — first name;
 - lastName** — last name;
 - sex** — gender, can be **MALE**, **FEMALE** or **NOT_DEFINED**;
 - country** — country;

language — language;
birthday — birthday.

There are other fields in the account structure, but it's not obligatory to present all fields, but you are welcome to do so.

The UI

The UI must present the given data. It must lazy load the data while scrolling. It must present correct groups order based on **groupOrder** item's property. Sorting in groups must be done in alphabetical order. The UI should allow to filter data based on a user's input.

The UI item of the list must have the following elements: **avatar**, **firstName** and **lastName**.

The avatar must be generated base on **firstName**, **lastName** and **sex**. Its size is **32x32 pixels** and it should be a circle. **firstName** and **lastName** are used to get initials to render in the resulting avatar. The color of the avatar is based on **sex's** value:

FEMALE — #FCD0FC;
MALE — #B5E6FF;
NOT_DEFINED — #E1E8ED.

Clicking on an item should open a dialog with detailed information about the contact. The dialog should show avatar of size **128x128 pixels**, **userName**, **firstName**, **lastName**, **sex**, **country**, **language** and **birthday**. How to layout widgets in the dialog, it's up to you.