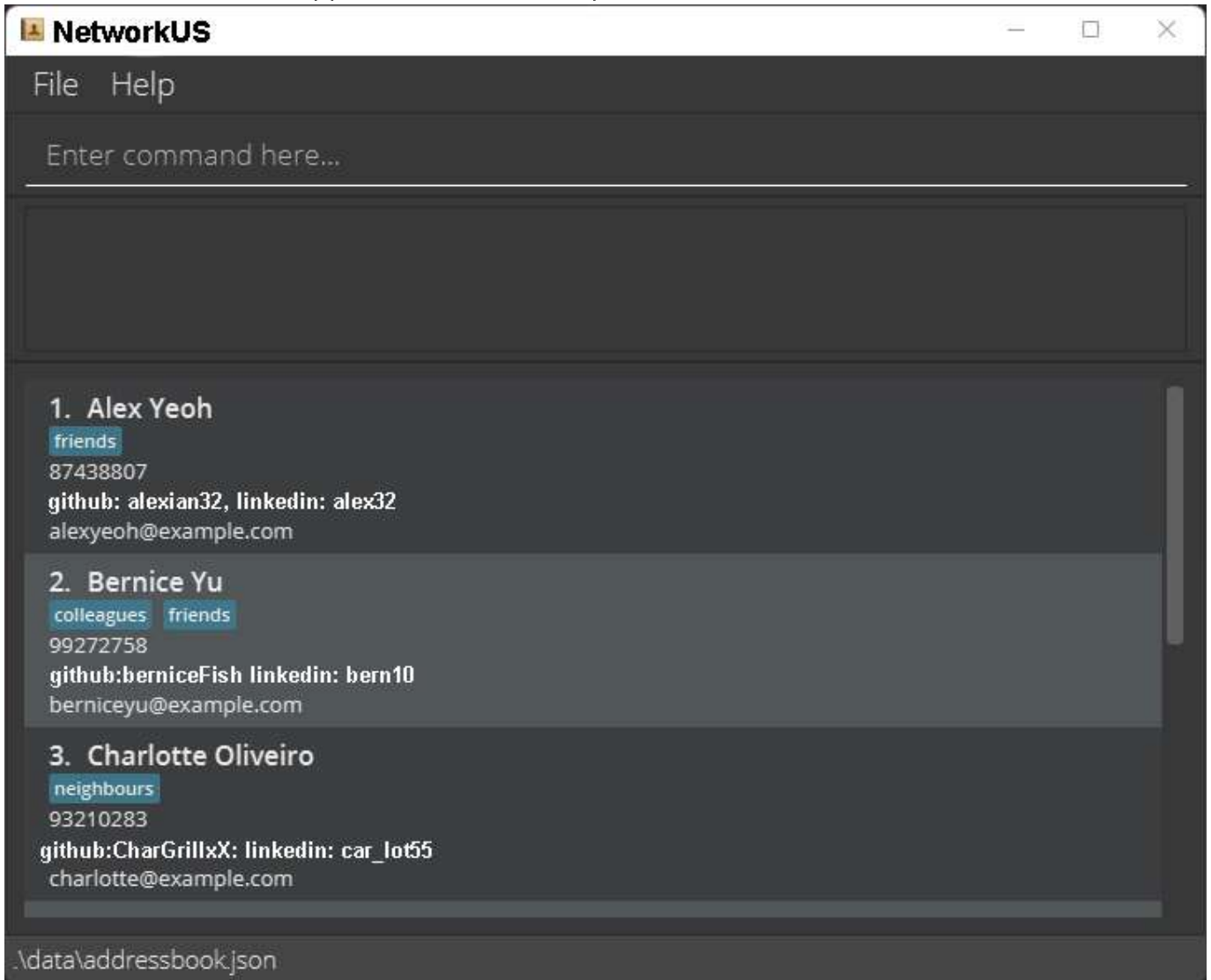# NetworkUS

User Guide

NetworkUs is a desktop app for organizing contacts. Its features assists in group management, allowing for quick searches of contacts with desired trailts (i.e specialization) and hence project group formation. It is catered for Computer Science Students. The app is optimized for the Command Line Interface (CLI) to allow swift access for keyboard users. The app is also complemented with Graphical User Interface (GUI) to display the program.

Quick start

1. Ensure you have Java `11` or above installed in your Computer.

2. Download the latest `networkus.jar` from here.

3. Copy the file to the folder you want to use as the *home folder* for your NetworkUS app.

4. Double-click the file to start the app. The GUI similar to the below should appear in a few seconds. Note how the app contains some sample data.

5. Type the command in the command box and press Enter to execute it. e.g. typing `help` and pressing Enter will open the help window.
   Some example commands you can try:

   - `list` : Lists all contacts.

   - `add` `n/John Doe p/98765432 e/johnd@example.com` : Adds a contact named `John Doe` to the NetworkUS.

   - `delete` `3` : Deletes the 3rd contact shown in the current list.

   - `clear` : Deletes all contacts.

   - `exit` : Exits the app.

6. Refer to the Features below for details of each command.

Features

> **ℹ Notes about the command format:**
>
> - Words in `UPPER_CASE` are the parameters to be supplied by the user.
>   e.g. in `add n/NAME`, `NAME` is a parameter which can be used as `add n/John Doe`.
>
> - Items in square brackets are optional.
>   e.g `n/NAME [t/TAG]` can be used as `n/John Doe t/friend` or as `n/John Doe`.
>
> - Items with `…` after them can be used multiple times including zero times.
>   e.g. `[t/TAG]…` can be used as ` ` (i.e. 0 times), `t/friend`, `t/friend t/family` etc.
>
> - Parameters can be in any order.
>   e.g. if the command specifies `n/NAME p/PHONE_NUMBER`, `p/PHONE_NUMBER n/NAME` is also acceptable.
>
> - If a parameter is expected only once in the command but you specified it multiple times, only the last occurrence of the parameter will be taken.
>   e.g. if you specify `p/12341234 p/56785678`, only `p/56785678` will be taken.
>
> - Extraneous parameters for commands that do not take in parameters (such as `help`, `list`, `exit` and `clear`) will be ignored.
>   e.g. if the command specifies `help 123`, it will be interpreted as `help`.
>
> - The detail field will be truncated if the content is too long to fit in the window.

Viewing help : `help`

Provides functionality to access the help page in the browser.

Format: `help`
Keyboard shortcut: `f1` then `enter`

Adding a person: `add`

Adds a person to the list of contacts.

Format: `add n/NAME [p/PHONE_NUMBER] [e/EMAIL] [g/GITHUB_USERNAME] [l/LINKEDIN] [d/DETAIL] [t/TAG]…`

> 💡 **Tip:** A person can have any number of tags (including 0)

Examples:

- `add n/Johnny Sim p/91234567`
- `add n/Betsy Crowe t/friend e/betsycrowe@example.com p/1234567 t/criminal g/lovelyprisoner`

Listing all persons : `list`

Shows a list of all persons in the contact list and the total number of contacts.

Format: `list`

Editing a person : `edit`

Edits an existing person in the list of contact.

Format: `edit INDEX [n/NAME] [p/PHONE] [e/EMAIL] [g/GITHUB_USERNAME] [l/LINKEDIN] [d/DETAIL] [t/TAG]…`

- Edits the person at the specified `INDEX`. The index refers to the index number shown in the displayed person list. The index **must be a positive integer** 1, 2, 3, …
- At least one of the optional fields must be provided.
- Existing values will be updated to the input values.
- When editing tags, the existing tags of the person will not be removed i.e adding of tags is cumulative.
- You can remove all the person's tags by typing `t/` without specifying any tags after it. The same applies for description `d/` .

Examples:

- `edit 1 p/91234567 e/johndoe@example.com` Edits the phone number and email address of the 1st person to be `91234567` and `johndoe@example.com` respectively.
- `edit 2 n/Betsy Crower t/` Edits the name of the 2nd person to be `Betsy Crower` and clears all existing tags.

Finding persons: `find`

Finds persons whose names or tags contain any of the given keywords from the list of contacts.

Format: `find KEYWORD [MORE_KEYWORDS]`

- The command is not case-sensitive, `find john` will match `John`
- Currently, only the name and their tags are searched.
- Persons whose names or tags match at least one keyword will be returned. `find john smith` will return `John Fletcher` and `Smith George`

Examples:

- `find John` will return `john`, `John Doe` and `Aaron john`
- `find jonas aaron silva` will return `jonas`, `Aaron john` and `Ann Silva`
- `find friends` will return anyone whose names or tags have the word `friends` in them

Deleting a person : `delete`

Deletes the specified person from the NetworkUS contacts.

Format: `delete INDEX`

- Deletes the person at the specified `INDEX`.
- The index refers to the index number shown in the displayed person list.
- The index **must be a positive integer** 1, 2, 3, …

Examples:

- `list` followed by `delete 2` deletes the 2nd person in the contact list.
- `find Betsy` followed by `delete 1` deletes the 1st person in the results of the `find` command.

Tagging persons : `addtag`

Tags the specified persons from the NetworkUS contacts with the specified tag.

Format: `addtag INDEXES TAG`

- Tags the persons at the specified `INDEXES`.
- The index refers to the index number shown in the displayed person list.
- The indexes **must be unsigned integers** 1, 2, 3, ...
- Tags must be alphanumeric without spaces.
- Will tag persons upto an out-of-bounds index or all indexes.
- Will not add duplicate tags to persons.

Examples:

- `list` followed by `addtag 1 2 programmer` tags the 1st and 2nd persons in the contact list with a programmer tag.
- `find John` followed by `addtag 1 2 programmer` tags the 1st and 2nd persons in the results of the `find` command with a programmer tag.

Clearing all entries : `clear`

Clears all entries from the contact list.

Format: `clear`

Exiting the program : `exit`

Exits the program.

Format: `exit`

Saving the data

Contact list data are saved in the hard disk automatically after any command that changes the data. There is no need to save manually.

Editing the data file

Contacts data are saved as a JSON file `[JAR file location]/data/networkus.json` . Advanced users are welcome to update data directly by editing that data file.

> ❗ **Caution:** If your changes to the data file makes its format invalid, NetworkUS will discard all data and start with an empty data file at the next run.

FAQ

Q: How do I transfer my data to another Computer?
A: Install the app in the other computer and overwrite the empty data file it creates with the file that contains the data of your previous AddressBook home folder.

Command summary

Action	Format, Examples
Add	`add n/NAME [p/PHONE_NUMBER] [e/EMAIL] [g/GITHUB_USERNAME] [l/LINKEDIN] [d/DETAIL] [t/TAG]…` e.g., `add n/James Ho p/22224444 e/jamesho@example.com t/friend t/colleague`
Addtag	`addtag INDEXES TAG` e.g., `addtag 1 2 programmer`
Clear	`clear`
Delete	`delete INDEX` e.g., `delete 3`

Action	Format, Examples
Edit	`edit INDEX [n/NAME] [p/PHONE_NUMBER] [e/EMAIL] [g/GITHUB_USERNAME]` `[l/LINKEDIN] [d/DETAIL] [t/TAG]…` e.g., `edit 2 n/James Lee e/jameslee@example.com`
Find	`find KEYWORD [MORE_KEYWORDS]` e.g., `find James Jake`
List	`list`
Help	`help`