Qordoba CLI v4

Re-Building using Go programing language for mainly 5 reasons

Compiles to a single static binary

```
GOOS=windows go build -o cli.exe
GOOS=linux go build -o cli
GOARCH=armv7 GOOS=linux go build -o cli-rpi
```

Easy to create a REST client (Go includes a no-nonsense `http` client and has built-in support for working with `xml`, `json` and binary formats.)

Fast on every platform

Integrate with package managers

The standard Go library includes a `flags` package that can be used to parse flags or arguments in a couple of lines of code. (Go is unpretentious - you can have a single file that compiles to a tiny binary and is done with that.)

Command Reference

The CLI should only use the application token

1. ### Push files (Support file versions)

```
qor push
Push all local source files to the Qordoba project.

*files
 (Optional). Lists the file paths to upload.
--version
 (Optional). Sets the version tag. Updates the file with version tag. Uploads a  file with the
version tag

--file-path pushs entire (relative) file paths. Please review the commands documentation further
down.

Update: the file will be updated if the file name or version already exists
```

2. ### qor download files (Support file versions - support original source file)

   Default file download command will give you two things A)only the completed files B) will give you all the files (all locals and audiences without source file)

   -c --current to pull the current state of the files
   -a --audiences Option to work only on specific (comma-separated) languages. example: `qor pull -a en-us, de-de`

   -s --source file option to download the update source file

   -o original file option to download the original file (note if the customer using -s and -o in the same command rename the file original to filename-original.xxx)
   --skip skip downloading if file exists.

3. ### Delete (Support file versions)

```
$ qor delete file_name.doc --version 1
```

4. **ls (show 50 only)**

```
+--------+-----------------+-----------+-----------------------------------------------+---------+
| ID     | NAME            | version   |  tag       | #SEGMENTS | UPDATED_ON          | STATUS  |
+--------+-----------------+-----------+------------------------+-----------------------------+
| 728068 | runtime.yaml    |           |   dev      |    100    | 2017-01-20 10:19:27 | Enabled |
| 725570 | TestFileMH.json | 2         |  test      |    300    |2016-12-21 22:40:13  | Enabled |
+--------+-----------------+-----------+------------------------+-----------------------------+
```

5. **Status per project or file (Support file versions)**

```
$ qor status
```

```
+----------+--------+-----------+---------+--------------+-----------+
| #AUDIENSES | #WORDS | #SEGMENTS | EDITING | PROOFREADING | COMPLETED |
+----------+--------+-----------+---------+--------------+-----------+
| ja-jp    | 94     | 32        | 0%      | 78.13%       | 21.88%    |
| es-es    | 94     | 32        | 100%    | 0%           | 0%        |
+----------+--------+-----------+---------+--------------+-----------+
```

$ qor status --json

{
"command":
Unknown macro: { "value"}

,
"audienses_id": 1,
"audienses_code": "A green door",
"audienses_namer": "A green door",
"audienses_code": "A green door",
"audienses_code": "A green door",
"audienses_code": "A green door",
"audienses_name": 12.5
}

```
$ qor status file_name.docx --version
+----------------+--------+-----------+---------+--------------+-----------+
| FILE NAME      | #WORDS | #SEGMENTS | EDITING | PROOFREADING | COMPLETED |
+----------------+--------+-----------+---------+--------------+-----------+
| file_name.docx | 94 | 32 | 0% | 78.13% | 21.88% |
+----------------+--------+-----------+---------+--------------+-----------+
```

$ qor status file_name.docx --json

{
"command":
Unknown macro: { "value"}

,
"audienses_id": 23,
"audienses_code": "en-US",
"filename": "file_name.docx",
"file_id": 123442,
"words": 94,
"segments": 32,
"status": "completed"
}

6. Init
7. **Add key**

$ qor add-key file_name.doc --version v1 --key "/go_nav_menu" --value "text text text" --ref "Main nav text"

8. **Update value by key**

```
$ qor update-value file_name.doc --version v1 --key "/go_nav_menu" --value
"text text text" --ref "Main nav text"
```

9. **Delete key**

```
$ qor delete-key file_name.doc --version v1 --key "/go_nav_menu"
```

10. **Create content (ph2 - later - not now)**

```
$ qor new file_name
> Text editore........
```

11. **Pull value by key**

```
$ qor value-key file_name.doc --version v1 --key "/go_nav_menu"
>
+---------------+---------------+-------------+---------------+--------------+-------------------------+
| FILE NAME | #VERSION|#KEY | #VALUE | #REF | #TIMESTAMP                          |
+---------------+---------+-------------+---------------+--------------+-------------------------+
| file_name.docx | v1. | /go_nav_menu | text text text | Main nav text | 01/23/2019 @ 9:59pm (UTC)
|
+---------------+---------+-------------+---------------+--------------+-------------------------+
```

12. **Score per file (ph2 - later - not now)**

```
$ qor score file_name.doc --version v1
```

13. **Results as JSON format**

    Adding --JSON front of any command will give you the response as JSON

14. **Help**

```
$ qor -h
$ qor -help
```

15. **Version**

$ qor -v
$ qor -version
>Qordoba Cli v4.0

CLI config

1. Local & file extensions code mapping support between source and target

```
qordoba:
access_token:
organization_id: 1234
workspaces_id: 12321
audiences_map: [ "123=EN_us", "12=EN_uK", "45=frFFrfs" ]

push:
sources:
- file: "./.lproj/Localizable.strings"

downloed:
targets:
- file: "./<audience_code>/<filename>.<extension>" =>> Waseem will review the options with the
customers
- rename_folder_in_path: ["folder_name=XXXXX"] Or ["folder_name=<audience_code>"]

blacklist:
sources:
- file: "./<audience_code>/<filename>.<extension>"
```

Reading the config files

Read configuration from `~/.qordoba/config-v4.yaml`

Check if current folder contains ./.qordoba.yaml if not search a parent directories for one.

If you find `.qordoba.yaml` set directory with this file as a root to the plugin operations.

Read content of the `.qordoba.yaml` overrides whatever is in `~/.qordoba/config-v4.yaml`