Crowdin

This library is using *Crowdin API* to perform the following operations
againts *travel_management_platform* translation files:

- *download*
- *sync*
- *update*

Installation

Add this line to your application's Gemfile:

```
gem 'crowdin'
```

And then execute:

```
$ bundle
```

Or install it yourself as:

```
$ gem install crowdin
```

Usage

Project initialization

Adding folders to the remote project

```
bundle exec rake crowdin:translations:add_dirs[locale]
```

Adding files to the remote project

```
bundle exec rake crowdin:translations:add_files[locale]
```

Setting up the remote project with a single command

```
bundle exec rake crowdin:translations:setup[locales]
```

> *Warning: These commands must be run only once (at the moment of creation). To avoid translation progress losses better use:* `sync` *and* `update_files` *when remote project is already initialized*

> *Warning:* `setup` *command does the following in background:*

- removes **/download** folder if present
- removes remote folders
- adds remote folders
- adds remote files

Translations fetching

Export translations (*build latest changes on Crowdin*)

```
bundle exec rake crowdin:translations:export[locale]
```

Download translations

```
bundle exec crowdin:translations:download[locale]
```

> Note: Run `export` command each time before `download`

> Note: `download` command will fetch the remote files, create a local folder **config/locales/download** and store the fetched files there. Files can be manually copied to **config/locales/<locale>/...**

Files synchronization

Synchronize folders

```
bundle exec rake crowdin:translations:sync_dirs[locale]
```

Synchronize files

```
bundle exec rake crowdin:translations:sync_files[locale]
```

Synchronize folders and files with a single command

```
bundle exec rake crowdin:translations:sync[locale]
```

Translation keys update

Update all remote files (*keeps translation progress*)

```
bundle exec rake crowdin:translations:update_files[locale
]
```

Update specific remote files

```
bundle exec rake crowdin:translations:update_files[locale
] -- --file <path to file>
```

> Note: Adds the missing keys to the remote project and keeps the
> **Crowdin** translation progress

Translations status

```
bundle exec rake crowdin:translations:status[locale]
```

> Note: This command will return the information about translation
> progress

Example of usage

Starting point. Phase 1

- **Initialization**
  - 1st option

    ```
    bundle exec rake crowdin:translations:add_dir
    s[locale]
    ```

    ```
    bundle exec rake crowdin:translations:add_fil
    es[locale]
    ```

  - 2nd option

    ```
    bundle exec rake crowdin:translations:setup[l
    ocale]
    ```

Phase 2

Working with translations is a continuous process. It is a matter of time when some member of the development team will add a new translation file or a new translation key to an existing file, or even both of them at the same time. Hence one needs to synchronize periodically the local files with the remote ones so that the *Crowdin* translator sees the new files/strings and performs the translation process.

```
bundle exec rake crowdin:translations:sync[locale]
```

> *Note: The* `sync` *and* `update_files` *commands must be used to let the translator keep track of the new translation files and keys.*

> *Note: The* `status` *command may be used to keep track of translator's progress.*

Phase 3

This is not actually a phase. Whenever is needed one can run the `export` and `download` commands and fetch the translations from **Crowdin**.

What can be improved?

- Add validation for `add_dirs`, `add_files` and `setup` commands. Skip the execution if remote project had been already initialized

Development

After checking out the repo, run `bin/setup` to install dependencies. Then, run `rake spec` to run the tests. You can also run `bin/console` for an interactive prompt that will allow you to experiment.

To install this gem onto your local machine, run `bundle exec rake install`. To release a new version, update the version number in `version.rb`.

Contributing

Bug reports and pull requests are welcome on GitHub at

https://github.com/locomote/crowdin.

License

The gem is available as open source under the terms of the MIT
License.