Automated Testing

This app includes predefined tests in the file `tests.py`, allowing for automated testing with an arbitrary number of “bots” simulating participant behavior. The bots for this app are programmed to behave consistent with theory but randomly, i.e. they choose “Option A” or “Option B” at random but do alternate between “Option A” and “Option B” at most once.

To run a test on the particular parameterization of the app defined in `config.py`, make sure to include a session configuration in `SESSION_CONFIG` in `settings.py`.

Open your terminal and run the command `otree test <name> <n>` from the project’s root directory, where `<name>` refers to the name of the session configuration and `<n>` refers to the number of bots to be simulated simultaneously. In addition, the `--export` flag can be added to export the data generated by the bots to a `*.csv-file` (stored in *oTree*’s project directory). For example, the command `otree test mpl 50 --export` will simulate a session of 50 participants for the session named `mpl` and export the generated data to a spreadsheet table.

To test the apps in a more full and realistic way, tests can be launched using *oTree*’s “browser bots'', simulating the app in an actual web browser. To do so, add `‘use_browser_bots’: True` to the respective session configuration in `settings.py`, run the server and create a session. The pages will auto-play with browser bots once the start links are accessed. Alternatively, browser bots can be started from the terminal by running the server (`otree runserver`) first and calling the command `otree browser_bots <name> <n>`, which will result in auto-play of `<n>` browser bots in Google Chrome©.[1]

For more detailed information on automated tests in *oTree*, refer to *oTree’s Docs.*

[1] To use a browser other than Google Chrome©, add the setting `BROWSER_COMMAND` to `settings.py`, specifying the full path to the executable of the desired web browser.