

Experiments with oTree and Python

Moinul Islam

9/17/2022

Table of contents

Preface	3
1 Operationalize oTree	4
1.1 App	4
1.1.1 game_app	4
1.2 Models.py	5
1.3 Pages.py	5
2 Chapter 2	7
3 Chapter 3	8
4 Chapter 4	9
5 Chapter 5	10
6 Chapter 6	11
7 Chapter 7	12
8 Chapter 8	13
9 Chapter 9	14
10 Chapter 10	15
References	16

Preface

oTree is an open-source and online software for implementing interactive experiments in the laboratory, online, the field or combinations thereof. The basic experimental setup in oTree consists of (i) an experiment written within oTree, (ii) a server computer, which can be a cloud server or a local laptop and (iii) subjects' devices with a web browser. oTree creates an experimental session on the server, as well as links for all the participants and the experimenter.

oTree is a framework based on Python that lets you build:

- Multiplayer strategy games, like the prisoner's dilemma, public goods game, and auctions
- Controlled behavioral experiments in economics, psychology, and related fields
- Surveys and quizzes

1 Operationalize oTree

1.1 App

1.1.1 **game_app**

1. To create an application named **game_app** move to the oTree folder

- `cd oTree`

2. Create the application

- `otree startapp game_app`

3. Move to the folder **game_app**

4. In this folder, you will find the following files as default

- `models.py`
- `pages.py`
- `tests.py`

5. In this folder, you will also find a subfolder

- `templates/game_app`
 - `Mypage.html`
 - `Results.html`

1.2 Models.py

A model is basically a database. Here we define the structure of the data. For instance, in a three data models. This is python **class**

- Subsession
 - Group
 - Player
1. class Subsession(BaseSubsession):
 - pass
 2. class Group(BaseGroup):
 - pass
 3. class Player(BasePlayer):
 - pass

1.3 Pages.py

- Pages that the participants see are defined in pages.py
 - Logic for how to display the HTML templates
 - when, how, and what to display
- page_sequence gives the order of pages
 - If there are multiple rounds the sequence is repeated

For instance,

1. class MyPage(Page):
 - pass
2. class ResultsWaitPage(WaitPage):
 - def after__all__players__arrive(self):
 - pass
3. class Results(Page):
 - pass

- `page_sequence = [MyPage, ResultsWaitPage, Results]`

2 Chapter 2

3 Chapter 3

4 Chapter 4

5 Chapter 5

6 Chapter 6

7 Chapter 7

8 Chapter 8

9 Chapter 9

10 Chapter 10

References