

Workshop: Section 2 - Basic Structure of oTree Experiments (Models, Pages, Templates)

Philipp Chapkovski
University of Bonn
chapkovski@uni-bonn.de

September 11th - 12th, 2023

Introduction to oTree's Architecture

- **Main components of any oTree project:**
 - Apps (+settings)
 - Models
 - Pages
 - Templates

Models in oTree

- Use to define data to store at each level
- There are five (nested) models:
- Session:
 - Subsession
- Participant:
 - Player
- Group
- You can define new **fields** at Subsession, Group, and Player level
- You can store the data in `vars` at Session and Participant levels

Pages in oTree

- A page (in z-Tree term, screen) to show consists of 3 elements:
- Page class
- position of the Page in the sequence of pages (`page_sequence`)
- the html (text) to show in the `Page.html`

Templates in oTree

- within html of the page you can use:
- plain html (`hello, I am bold!` will result in: **hello, I am bold!**)
- CSS styles (look at the [Bootstrap docs](#) for details)
- JavaScript
- and oTree own (kinda) template language to render at the server side (retrieving data)

Forms and User Input

- Storing data at the model level
- Data validation:
 - Static
 - Dynamic
- Getting data from user:
- Defining `form_model` and `form_fields` at the page level
- Showing them at the specific place of the html page using `{{formfields}}`

Displaying Data to Users

- Showing static (Constants) and dynamic (from other users/same user) data
- Showing data conditionally (`{{if-else}}` structures) and in arrays (`{{for}}`)
- Custom data formatting for a template (using `vars_for_template` and `js_vars`)

Concept of Apps in oTree

- An app is a set of models, pages, and templates.
- Specific configuration in `settings.py` can group several apps together or launch the same app (or set of apps) with different parameters.
- This can be used for between-session treatment assignment

Exercise: Creating Your First oTree Project

- Check if otree is installed: `otree version`
- Create an empty project: `otree startproject YOURPROJECTNAME`
(choose 'No' when asked whether to add sample games!)
- Move to a newly created folder: `cd YOURPROJECTNAME`
- Create a new app: `otree startapp YOURAPP`
- Register app in the `settings.py`:

```
dict(  
    name='public_goods',  
    app_sequence=['YOURAPP'],  
    num_demo_participants=3,  
)
```

- Launch the server: `otree devserver`
 - Sometimes right after installation oTree can ask to delete the temporary database file first (`db.sqlite3`) - just delete it.
- Go to the local server in your browser: <https://localhost:8000>