CREED Experiment Setup Guide

Updated May 2025
Tell Julian T. if anything is wrong or missing and we can update it!

Contents

1 Experimental Timeline

1.1 Overview

Before running an experiment at the CREED Lab or applying for ACBC funding, present your project at the CREED Lunch Seminar. Contact Julian Kirschner or Madgalena Wasilewska to schedule. Complete the following steps:

1. **Pre-analysis plan:** AsPredicted

2. **Funding application:** ACBC

3. **UvA/EB Project Overview:** Research Management Services

   - Ethics review
   - Data Management Plan

Pre-Analysis Plan Questions

1. Has any data already been collected?

2. What is the main question or hypothesis?

3. Key dependent variable(s) and measurement method?

4. Number and nature of conditions?

5. Planned analyses?

6. Outlier handling and exclusion criteria?

7. Sample size or rule?

8. Additional info?

2 Using the CREED Lab

2.1 Reserving the Lab

Coordinate with Joep to reserve lab access and receive credentials for CREEDExperiment.nl.

2.2 Session Management

- Login details: from Joep
- Experiment number: from Ailko
- URL: `http://www.creedexperiment.nl/proftulp/loginexp.html`
- Create a new session via "new exp"

2.3   Managing Sessions

- Modify session: `Show Session > Modify`

- Send reminder: `Show Session > Reminder`

- Finish session: `Show Session > Finish`

2.4   Inviting Participants

1. Create table with name and email.

2. Upload table to invite pool.

3. Prepare a customisable email template.

4. Limit email sends (e.g., 100 per batch).

5. System tracks sent emails and avoids duplicates.

2.5   Sending Reminders

Navigate to Show Sessions, select a session, and click `Reminder`.

Tips:

- You can customise the reminder (subject, message).

- Add your email to test delivery.

- Double-check the session before sending.

3   Running in the Lab (Large lab)

If you're running in the other labs you will have to use a different postgreSQL as this one only exists in the large lab, and the URLs won't be the same as it is run on a different server.

3.1   Configure OTree

Set this in `settings.py`:

```
ROOMS = [
    dict(
        name='experiment',
        display_name='Experimental␣Session',
    ),
]

# Admin credentials from environment
import os
ADMIN_USERNAME = os.environ.get('OTREE_ADMIN_USERNAME')
ADMIN_PASSWORD = os.environ.get('OTREE_ADMIN_PASSWORD')

DATABASES = {
    'default': {
```

```
        'ENGINE': 'django.db.backends.postgresql_psycopg2',
        'NAME': os.environ.get('DB_NAME'),
        'USER': os.environ.get('DB_USER'),
        'PASSWORD': os.environ.get('DB_PASSWORD'),
        'HOST': os.environ.get('DB_HOST', 'localhost'),
        'PORT': os.environ.get('DB_PORT', '5432'),
    }
}
```

Listing 1: settings.py configuration

3.2 Adopt Player Labels

```
class Player(BasePlayer):
    participant_label = models.StringField()

    def set_participant_label(self):
        self.participant_label = self.participant.label
```

Listing 2: First app init.py

3.3 Start Session Script

Use a '.bat' script to automate environment setup:

```
@echo off
REM === Environment Setup ===
set DB_NAME=mistakesexperiment
set DB_USER=moi
set DB_PASSWORD=ExperimentAllDay
set DB_HOST=localhost
set DB_PORT=5432
set DATABASE_URL=postgres://%DB_USER%:%DB_PASSWORD%@%DB_HOST%:%DB_PORT%/%DB_NAME%

set OTREE_ADMIN_USERNAME=admin
set OTREE_ADMIN_PASSWORD=1234
set OTREE_PRODUCTION=1
set OTREE_AUTH_LEVEL=STUDY

cd "C:\Users\Admin\Desktop\TAIT\ExpMistakes Copy"
otree resetdb
start http://145.18.178.130:8000/rooms
otree prodserver
```

Listing 3: start_experiment.bat

3.4 Monitor the Session

Visit `http://145.18.178.130:8000/rooms` or use the shortcut on lab PCs.

4 PostgreSQL Setup

Note: oTree defaults to SQLite. For production, use PostgreSQL.

```
CREATE DATABASE experimentname;
CREATE USER moi WITH PASSWORD '1234';
GRANT ALL PRIVILEGES ON DATABASE experimentname TO moi;
```

Listing 4: psql setup

Then set the 'DATABASE_URL' environment variable accordingly.

5 Payment Handling

5.1 Cash

Ask your supervisor to email Ailko (ailko@xs4all.nl) with:

- Experiment number
- Total cash needed
- Preferred denominations

5.2 Digital Transfer

Collect IBANs from participants and forward to central administration.

6 oTree/Heroku FAQ

Q: How many participants can Heroku handle at once?

Based on experience, standard setups can support around **20–50 participants** simultaneously. Going beyond this increases the risk of server crashes.

Q: How can I scale to more participants (e.g., 80–120)?

A setup that successfully handled up to 120 participants used the following Heroku configuration:

- **Heroku Postgres Standard 0** ($50/month)
- **Heroku Redis Premium 2** ($60/month)
- **Heroku Dynos:**
  - Web dynos: 2X with 4 dynos ($200/month)
  - Worker dynos: 2X with 2 dynos ($100/month)

Tip: Dynos and Redis can be scaled up before a session and down afterward to reduce costs. However, the Postgres database must remain active to retain session data.

Q: Wait pages don't refresh automatically. What should I do?

If you encounter this issue, make sure to run:

```
heroku config:set PGSSLMODE=require
```

Listing 5: Heroku SSL Configuration

Without this, you may get the error: `FATAL: no pg_hba.conf entry for host`, and wait pages won't auto-update.

Q: Is there a workaround to prevent stuck wait pages?

Yes. You can add a fallback script that reloads the page every 10 seconds to avoid session break-downs:

```
setInterval(function(){ location.reload(); }, 10000);
```

Listing 6: Auto-refresh fallback

Note: This helped recover from issues during high-load sessions where the auto-update failed.