Deploying an oTree experiment using oTree Hub, alongside Heroku, ensures a smooth process with extra features for management and data collection. This step-by-step guide will walk you through from preparing your experiment in PyCharm to deploying it on oTree Hub with Heroku hosting.

**1. Prepare Your oTree Experiment in PyCharm**

a. Development and Testing

* **Develop in PyCharm**: Finalize the coding and local testing of your oTree experiment using PyCharm. Ensure all functionalities work as expected.
* **Dependencies**: Check your **requirements.txt** file to list all necessary dependencies, including the specific version of oTree and any other packages used.

b. Configuration

* **Settings Adjustment**: In your **settings.py**, set **DEBUG** to **False** for production and configure **ALLOWED\_HOSTS** with your future oTree Hub and Heroku URLs.

c. Final Local Test

* Run **otree devserver** in your terminal or command line to do a last check on your project's functionality.

**2. Version Control with Git**

a. Git Initialization

* If not already done, initialize a Git repository in your project's root folder with **git init**.
* Add all files to Git (**git add .**) and make your initial commit (**git commit -m "Initial commit"**).

b. GitHub Repository

* **Optional**: Create a GitHub repository for better version control and backup. Push your code using **git remote add origin YOUR\_GITHUB\_REPO\_URL** and **git push -u origin master**.

**3. Setting Up on oTree Hub**

a. Account Creation and Setup

* Sign up or log in to [oTree Hub](https://otreehub.com/).
* Navigate to the section where you can create or import your project.

b. Importing Your Project

* You can import your project directly from GitHub if you've pushed your code there. Alternatively, you can configure deployment through Git URLs provided by oTree Hub.

c. Configuration on oTree Hub

* Follow the oTree Hub instructions to set up your project. This includes configuring experiment settings, data collection preferences, and other deployment specifics.

**4. Deploying to Heroku Through oTree Hub**

a. Heroku Account and CLI

* Ensure you have a Heroku account and the Heroku CLI installed.
* Log into Heroku through the CLI (**heroku login**).

b. Creating and Configuring Heroku App

* In most cases, oTree Hub will guide you through creating a Heroku app directly from its interface.
* Follow the on-screen instructions to link your oTree project with a new or existing Heroku app.

c. Deployment

* oTree Hub provides a seamless integration with Heroku, often handling the deployment process behind the scenes. Follow any specific instructions provided by oTree Hub to deploy your app.

d. Database and Environment

* If your experiment requires a database, oTree Hub and Heroku together will typically manage this setup. Ensure any required environment variables or configurations are correctly set in the Heroku dashboard.

**5. Testing Your Deployment**

a. Access Your Experiment

* Once deployed, access your experiment via the URL provided by Heroku or the link in your oTree Hub project page.

b. Debugging

* Use **heroku logs --tail** in the CLI for real-time logs if you encounter any issues. oTree Hub may also offer tools for monitoring and troubleshooting.

**6. Final Steps**

a. Participant Instructions

* Prepare clear, concise instructions for your participants. Ensure they know how to access and navigate your experiment.

b. Launch

* Conduct any final tests to ensure reliability, then officially launch your experiment to your participants.

c. Monitoring and Data Collection

* Utilize oTree Hub and Heroku's monitoring tools to oversee your experiment's progress. Collect data as per your setup instructions on oTree Hub.

By following these detailed steps, you leverage oTree Hub's powerful features for your experiment, from streamlined deployment on Heroku to enhanced data collection and management. Remember to review oTree Hub's documentation for any specific guidance or updates. Good luck with your deployment and research!