



Co-funded by the
Erasmus+ Programme
of the European Union

Shared modelling and simulation environment (SMSE)



CPNU proposal

1st Training School for tutors
September 8, 2021



Co-funded by the
Erasmus+ Programme
of the European Union

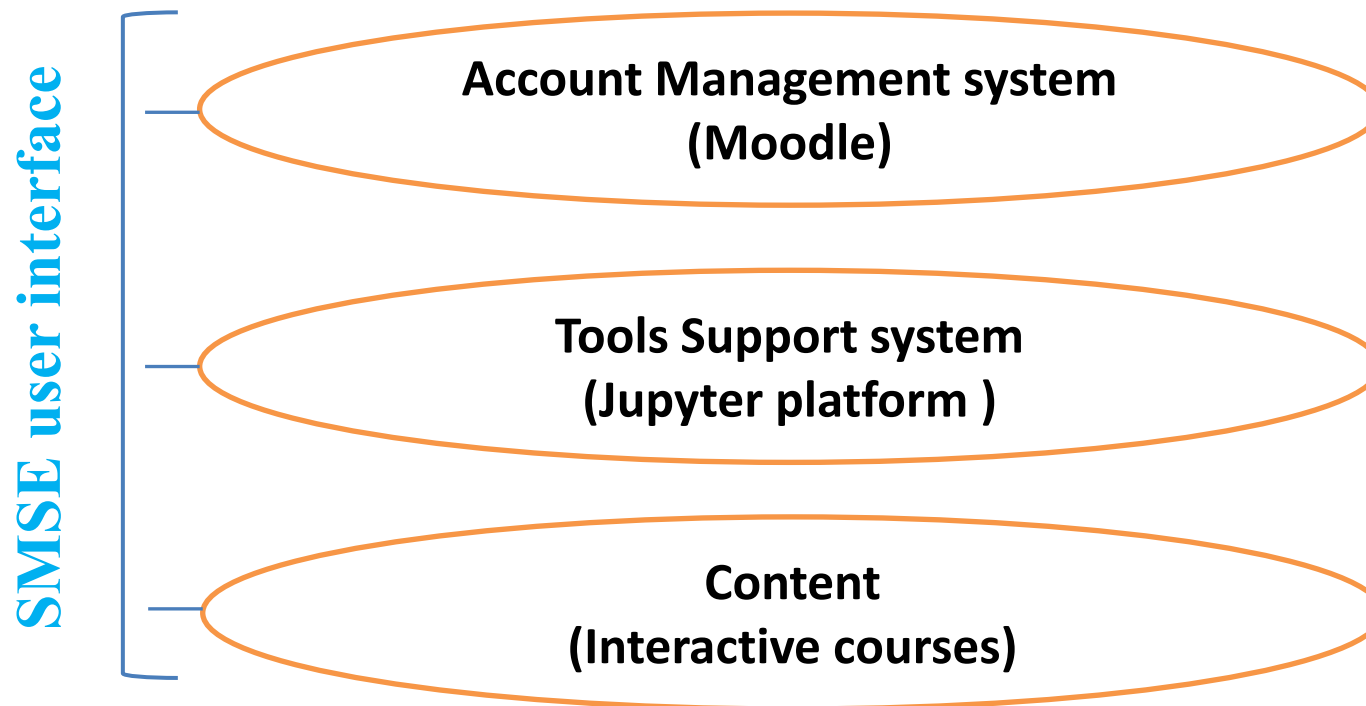
Background

- **CybPhys project commitments**
- **BSU SMSE suggestions and their discussions**
- **CPNU ICT Infrastructure**
- **Time and budget limitation**

1st Training School for tutors
September 8, 2021

SMSE architecture

Main idea and task – embedding Jupyter platform to Moodle

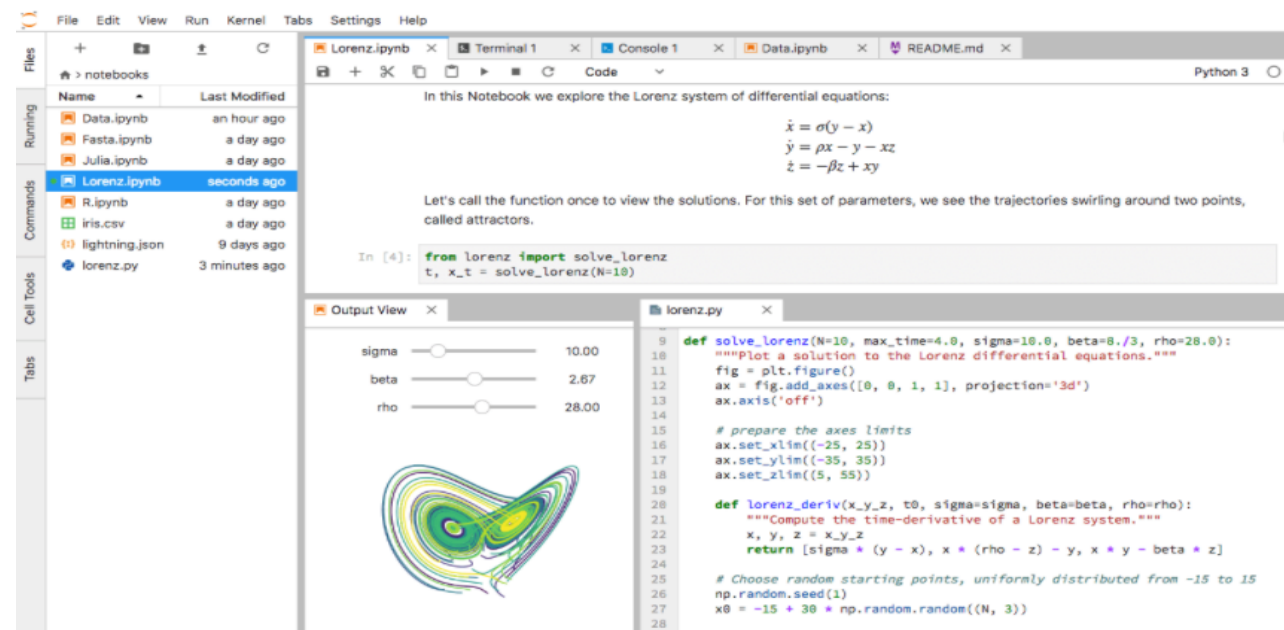


SMSE development goal

To provide users with the ability to create and use interactive courses in the form of Jupyter Notebook documents:

Multi-frame view with inserts of :

- Structured text
- Executed code parts
- Drawings and charts
- Results of modeling with plotting



File Edit View Run Kernel Tabs Settings Help

Lorenz.ipynb x Terminal 1 x Console 1 x Data.ipynb x README.md x Python 3

In this Notebook we explore the Lorenz system of differential equations:

$$\begin{aligned}\dot{x} &= \sigma(y - x) \\ \dot{y} &= \rho x - y - xz \\ \dot{z} &= -\beta z + xy\end{aligned}$$

Let's call the function once to view the solutions. For this set of parameters, we see the trajectories swirling around two points, called attractors.

In [4]: `from lorenz import solve_lorenz
t, x_t = solve_lorenz(N=10)`

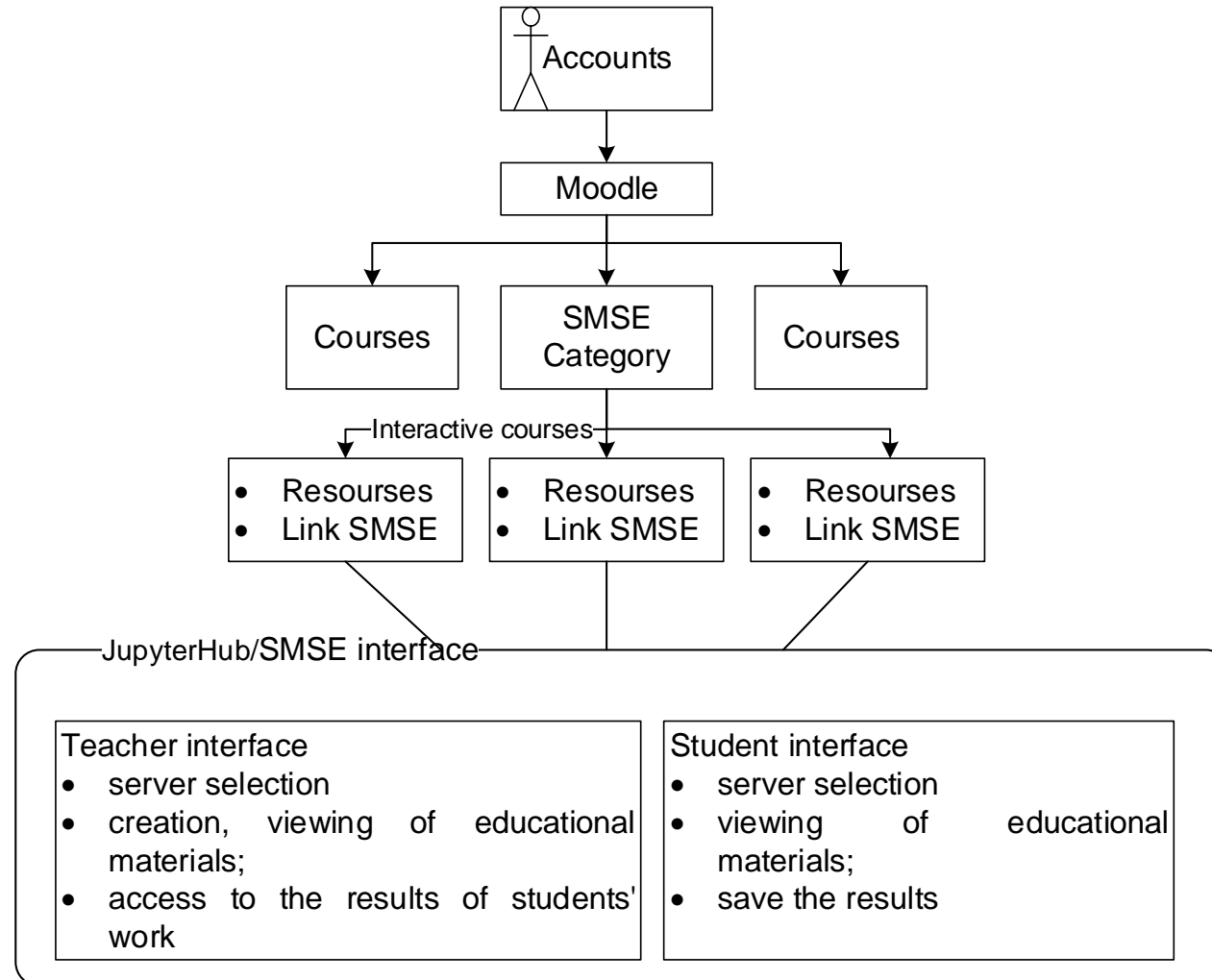
Output View x lorenz.py x

sigma 10.00
beta 2.67
rho 28.00

```
9 def solve_lorenz(N=10, max_time=4.0, sigma=10.0, beta=8./3, rho=28.0):
10     """Plot a solution to the Lorenz differential equations."""
11     fig = plt.figure()
12     ax = fig.add_axes([0, 0, 1, 1], projection='3d')
13     ax.axis('off')
14
15     # prepare the axes limits
16     ax.set_xlim((-25, 25))
17     ax.set_ylim((-35, 35))
18     ax.set_zlim((5, 55))
19
20     def lorenz_deriv(x,y,z, t0, sigma=sigma, beta=beta, rho=rho):
21         """Compute the time-derivative of a Lorenz system."""
22         x, y, z = x,y,z
23         return [sigma * (y - x), x * (rho - z) - y, x * y - beta * z]
24
25     # Choose random starting points, uniformly distributed from -15 to 15
26     np.random.seed(1)
27     x0 = -15 + 30 * np.random.random((N, 3))
28
```

1st Training School for tutors
September 8, 2021

SMSE use case diagram





Co-funded by the
Erasmus+ Programme
of the European Union

SMSE structure components

LMS Moodle :

- registration of SMSE users and control of them accounts
- integration of SMSE courses with another teaching materials and courses
- collection and visualization of test results and etc.

1st Training School for tutors
September 8, 2021

SMSE structure components

Jupyter platform:

- **JupyterHub** - to serve Jupyter notebooks for multiple users
- **JupyterLab** enables to work with Jupyter notebooks documents and its kernels (program languages, modelling tools etc.)
- **Jupyter Notebook** – final teaching document with possibility to run simple simulations and advanced analytical tools right from the browser



Co-funded by the
Erasmus+ Programme
of the European Union

SMSE structure components

LTI (Learning Tools Interoperability) authentication:

- Pass-through authentication with LMS Moodle
- Integration Moodle with Jupyter platform

1st Training School for tutors
September 8, 2021

SMSE structure components

Additional (optional) components:

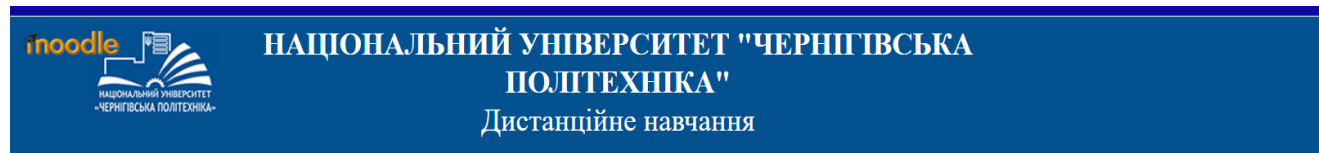
- **DockerSpawner (JupyterHub Docker Spawner)** enables JupyterHub to spawn single user notebook servers in Docker containers. Containers isolate software from its environment. It allows to develop a set of servers with various services for students.
- **Conda** - for installing Jupyter notebook kernels and switching easily between them
- **nbgrader** allows to easily create assignments for students that include both coding exercises, written responses and quickly grading completed assignments.

SMSE interface functional requirements

1. **Access to SMSE with Moodle account**
2. **Creation a SMSE environment with default kernels**
3. **Linking Jupyter notebook server with Moodle account**
4. **Adding kernels to Jupyter notebook server**
5. **Distributed access to Jupyter notebook documents and SMSE tools**
6. **Integration SMSE with Docker and nbgrade (optional)**
7. **Integration nbgrade with Moodle as for collection of completed assignments.(optional)**

Prototype of SMSE user interface

1. Login to Moodle of CPNU



vkazymyr

.....

Remember username

Log in


Forgotten your username or password?


Cookies must be enabled in your browser ?


Some courses may allow guest access

Log in as a guest

Log in using your account on:

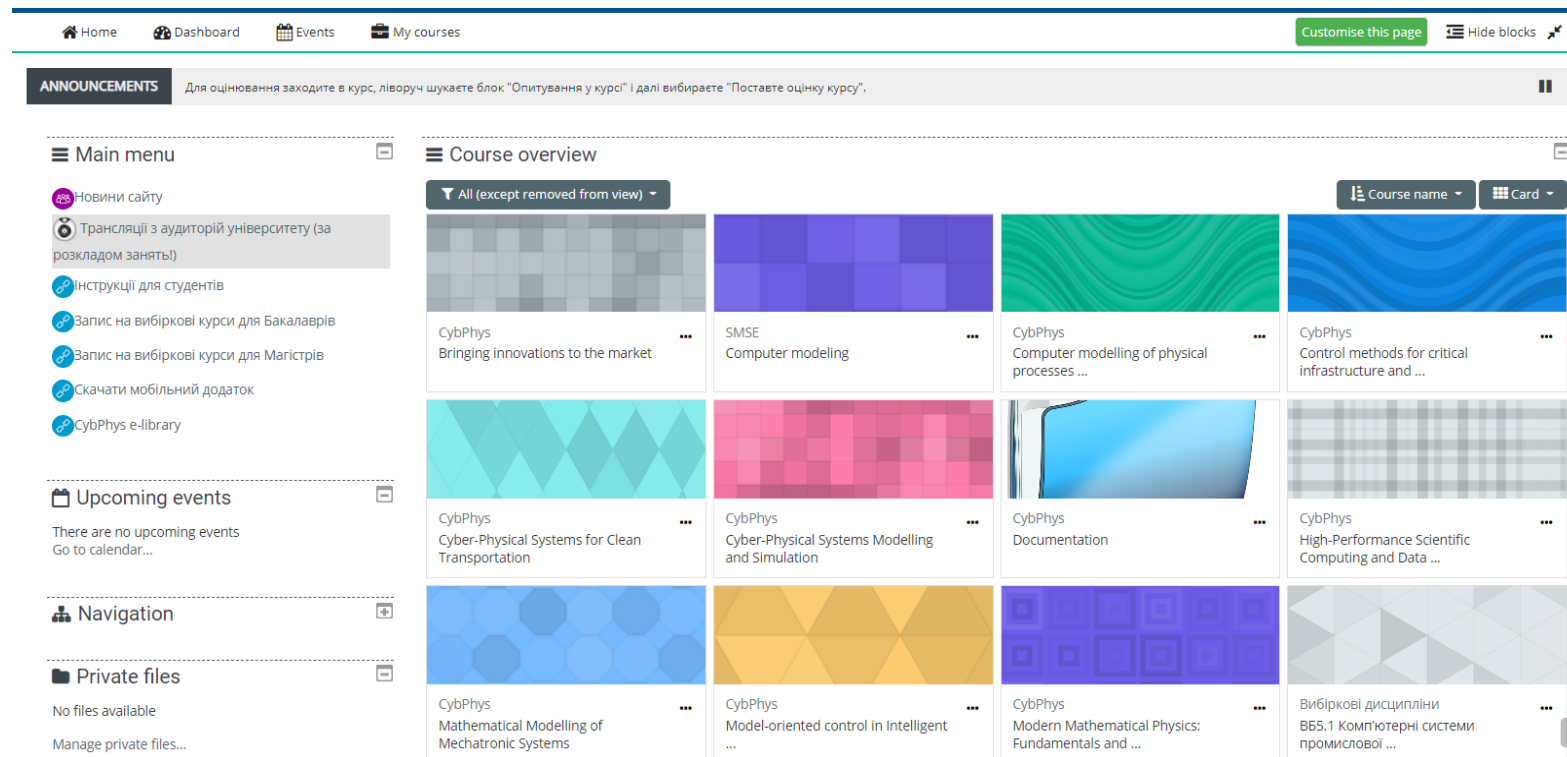
 Google

 Facebook

 Microsoft

Prototype of SMSE user interface

2. Select SMSE category on Moodle dashboard

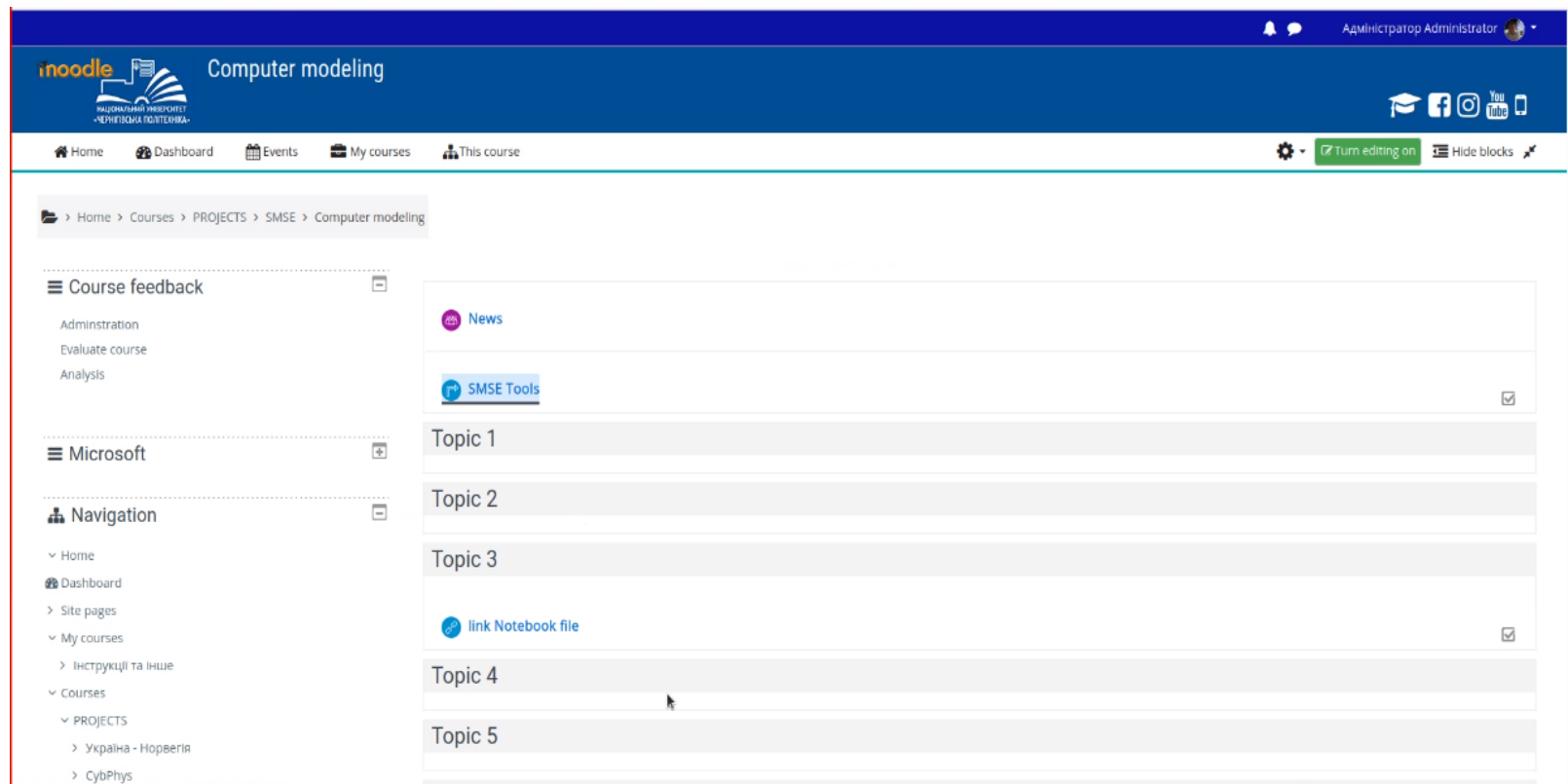


The screenshot shows a Moodle dashboard with a navigation bar at the top containing 'Home', 'Dashboard', 'Events', and 'My courses'. Below the navigation bar is an 'ANNOUNCEMENTS' section with a message about course evaluation. The main content area is divided into several sections:

- Main menu:** A list of links including 'Новини сайту', 'Трансляції з аудиторій університету (за розкладом занять)', 'Інструкції для студентів', 'Запис на вибіркові курси для Бакалаврів', 'Запис на вибіркові курси для Магістрів', 'Скачати мобільний додаток', and 'CybPhys e-library'.
- Upcoming events:** A section stating 'There are no upcoming events. Go to calendar...'.
- Navigation:** A section with a plus sign icon.
- Private files:** A section stating 'No files available. Manage private files...'.
- Course overview:** A grid of course cards. The 'SMSE Computer modeling' card is highlighted in purple. Other cards include 'CybPhys Bringing innovations to the market', 'CybPhys Computer modelling of physical processes...', 'CybPhys Control methods for critical infrastructure and...', 'CybPhys Cyber-Physical Systems for Clean Transportation', 'CybPhys Cyber-Physical Systems Modelling and Simulation', 'CybPhys Documentation', 'CybPhys High-Performance Scientific Computing and Data...', 'CybPhys Mathematical Modelling of Mechatronic Systems', 'CybPhys Model-oriented control in Intelligent...', 'CybPhys Modern Mathematical Physics: Fundamentals and...', and 'Вибіркові дисципліни ВБ5.1 Комп'ютерні системи промислової...'.

Prototype of SMSE user interface

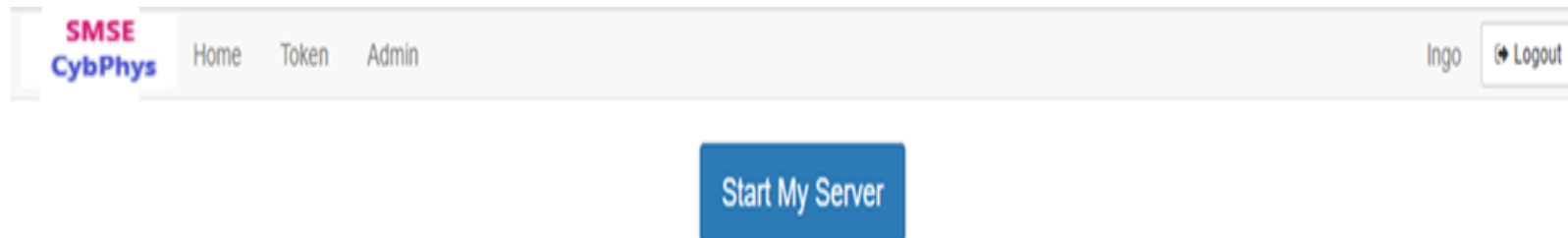
3. Start JupyterHub for SMSE - click on SMSE tools.



The screenshot displays the Moodle LMS interface for a course titled "Computer modeling". The breadcrumb trail indicates the path: Home > Courses > PROJECTS > SMSE > Computer modeling. The main content area shows a list of topics: "Topic 1", "Topic 2", "Topic 3", "Topic 4", and "Topic 5". A "SMSE Tools" block is visible, and a "link Notebook file" button is present. The left sidebar contains navigation menus for "Course feedback", "Microsoft", and "Navigation".

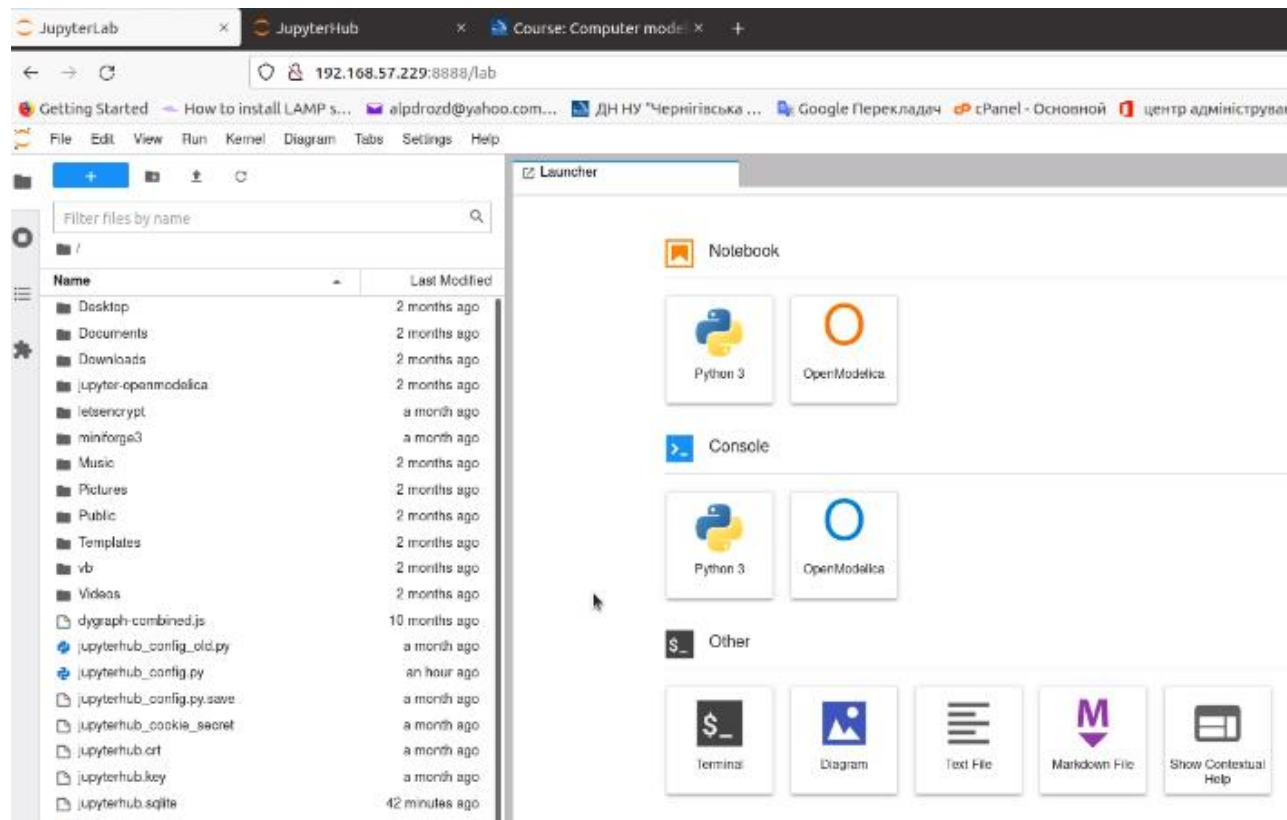
Prototype of SMSE user interface

4. Start JupyterLab



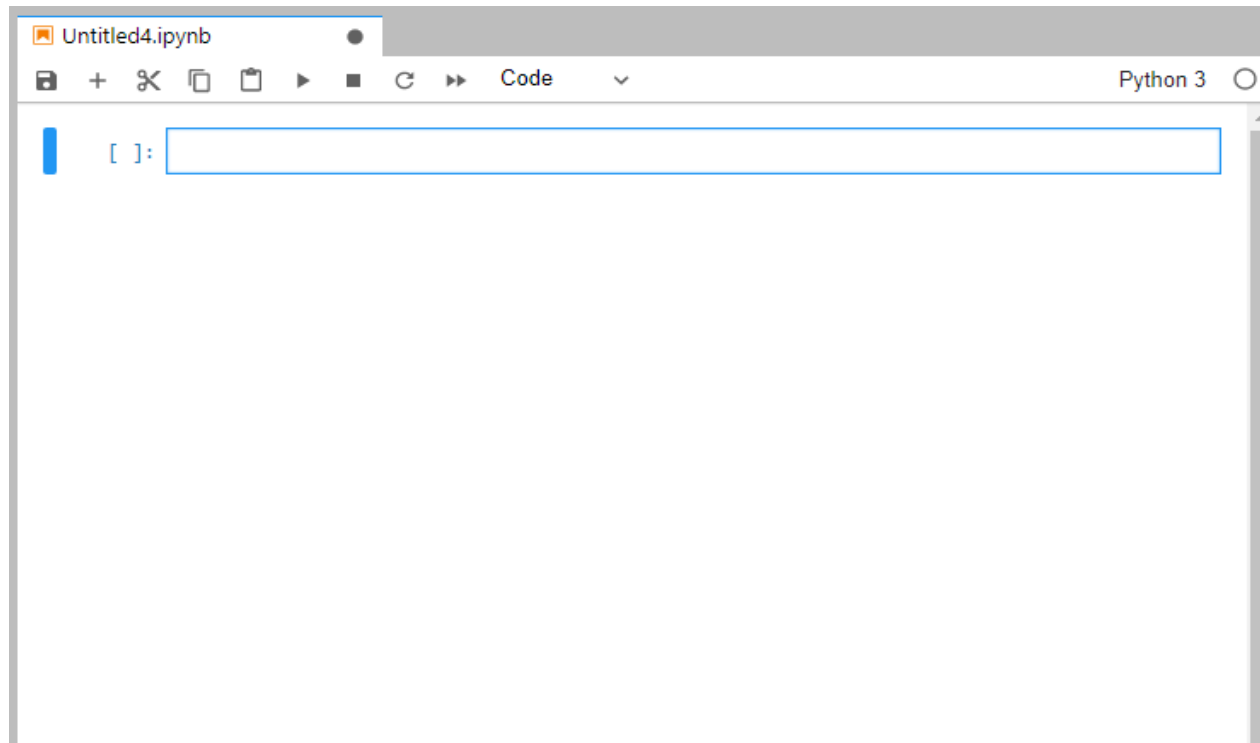
Prototype of SMSE user interface

5. Select document with Jupyter notebook kernel



Prototype of SMSE user interface

6. Select document will be opened (or created)



Hardware questions to be solved

1. Purchase of new more power Moodle server
2. Purchase of SMSE server (for deployment of Jupyter platform and SMSE user interface)
3. Upgrade of network

Software questions to be clarified

1. Creation the Jupyter notebook documents supported by several kernels
2. Integration nbgrade with Moodle
3. Using of clients kernels

Project plan

N	Activity	Term
1	Purchase of equipment (hardware and software)	October, 2021
2	Installation of the software on the servers of CPNU	November, 2021
	Configuring all software including account creation	December, 2021
	Implementation of the necessary functions	February, 2022
	Development of SMSE interface	March, 2022
	Development of SMSE course examples	May, 2022
	Testing of SMSE	June, 2022
	Development of documentation	July, 2022
	Delivery-acceptance of SMSE	August, 2022



Co-funded by the
Erasmus+ Programme
of the European Union

Project team

1. **Project manager**
2. **Programmer**
3. **System administrator**
4. **Tester**



Co-funded by the
Erasmus+ Programme
of the European Union

Contact information:

Chernihiv National University of Technology
14035, Ukraine, Chernihiv, Shevchenko Str., 95

site: www.stu.cn.ua

CybPhys project contacts:

Volodymyr Kazymyr

tel. +38 050 344 43 77

e-mail: vvkazymyr@gmail.com