Course on coursera from where you can get all the resources to study the concepts in this PESUIO course -

<https://www.coursera.org/learn/build-a-computer>

You can just directly submit the codes done in the PESUIO course and earn the certificate in the coursera course, you can also submit one of the projects given in the course link as your final project for this PESUIO course.

There are many projects you can do to submit as your final project, I would give you some of my ideas and standard projects, but I’d like you guys to try to come up with your own ideas and make a unique project in the scope of this course.

SOME PROJECTS YOU COULD DO -

1. Make an ALU for the hack computer -
   1. Goal - Make a fully functioning ALU for the hack computer given the truth table present in the edmodo under folder week 1 and simulate and test it using the Hardware simulator.
   2. To get an idea on how to make an ALU from the truth table given, you could refer to Week 2: Unit 2.4: Arithmetic and Logic Unit video on coursera.
   3. Make th ALU using the hack hdl, run and test the code using the simulator
   4. For presentation, run the code in front of the class and explain briefly the approach and the code itself, can also make a ppt if you want.
2. Make the CPU of the hack computer -
   1. Goal - Make a fully functioning CPU of the Hack computer, given the circuit diagram and figure out the appropriate inputs of the control bits of the A and C - instruction of the hack assembly language to the CPU of the computer.
   2. Please refer to week 4 of the course on coursera to understand the Hack assembly language and week 5 to understand the functioning of the CPU.
   3. Build a CPU in hdl in accordance to the diagram instructions given in the CPU folder (week 3) in edmodo.
   4. Test and run the hdl code in the simulator.
   5. For presentation, run the code in front of the class and explain briefly the approach and the code itself, can also make a ppt (optional).
3. Make the screen of the assembly language simulator show a letter when keyboard press is detected -
   1. Goal - To simulate the functioning of a key on the keyboard using hdl by giving appropriate response on the screen in the asm simulator.
   2. Please refer to week 4 of the course on coursera to understand the Hack assembly language.
   3. Pay special emphasis to input/output devices section, see how to detect keyboard input and manipulate the screen output.
   4. When a keyboard input is detected, make the screen provide a meaningful output (change of the screen pixel values) of your choice. Eg - screen shows ‘A’ when keyboard key ‘A’ is pressed.
   5. The output on the screen and input of keyboard can be of your choice.
   6. For presentation, run the code in front of the class and explain the input and corresponding change in output shown on your screen. Briefly explain the approach and the code itself, you can also make a ppt if you want.
4. Making an Assembler for the hack assembly language -
   1. Goal - To make an assembler for the hack assembly language using another programming language of your choice.
   2. Refer to the course on coursera week 4 to understand how to assembly language works
   3. Refer to week 6 of the same course to understand the algorithm to convert hack assembly language to binary.
   4. Give the input however you like, doesn’t need to be a file as in the course, show the correct binary output as a string of 0’s and 1’s.
   5. Run the output produced by your code through the simulator of the assembly language.
   6. Assume the input code given to translate doesn’t have any symbols or variables as in the hack assembly language. Only A and C instructions present.
   7. For presentation, run the code in front of the class and show the binary code output given by your program. Briefly explain the approach and the code itself (in the language of your choice), you can also make a ppt if you want.
5. You can choose any other project you want to make in the scope of this course, the projects above are just a few of my ideas. You could create and simulate a different chips, such as encoders and decoders, simulate an led display using the screen in the hdl simulator and so on. Please have a clear goal and and adequate documentation of your ideas.