Report on conducted simulation:

Simulation has been conducted in accordance with the settings described in the document containing the test task. I have set negative values for bias and depletion voltages, since it seems that such values determine the operation mode of mimosa26. Each simulation run took about 50s. The results were written in *gsoc_modules.root* file and later analysed via a TBrowser instance of ROOT framework. DetectorHistogrammer directory contains information about pixels hit by particles. From the *hit_map* diagrams one can conclude, that most of the particles passed almost through the centre of detectors, which meets the position settings of the detectors and the beam. The size of pixel clusters decreases with the sequence number of a detector following the beam direction. Thus, in the first detector most of clusters consisted of 4 pixels, and in the third detector – of 2 pixels. Other plots in the module directories correspond to the settings of simulations: for example, according to the diagrams, the digitizing threshold is at about 0.4 ke. Distribution of the drift time for charges in DUT is wider than in other two detectors – this is most likely caused by difference in electric field configurations.

I have also tested and played with other examples. For instance, I tried to set magnetic field at 20, 50, 100T just to play and watch electrons or positrons moving along ring and helix trajectories with small radii. Besides, I have run TCAS Field Simulation example with different numbers of workers from 1 to 4: the time consumption was about 33 minutes with 1 worker, about 21 minutes with 2 workers, and about 16 minutes for both configurations with 3 and 4 workers. I reckon, that there was almost no difference in performance for 3 and 4 workers, because I allocated only 4 cores for my virtual machine with Linux, were the framework is installed.

Additional thoughts and questions

1. It is rather difficult to check whether everything works as planned without understanding what actually has to be happening. Is there article or blog post about the context in which Allpix Squared is supposed to be employed? Another question is how to read results of simulations? In general, it seems to be quite clear, that the software allows one to simulate detection of particles by different detectors, doesn't it? Probably such understanding is enough for performing the actual task of the GSoC project.
2. By the way, I have run Capacitive Transfer example and observed diagrams via TBrowser of ROOT, but plots of separate pixels do not contain meaningful axis descriptions and titles, so one has to guess what is shown in those graphs.
3. Small question deriving, probably, from my lack of understanding of how detector actually work. What does "residual" mean in context of detectors: such plots are contained in DetectorHistogrammer directory.

Trouble and problems:

1. Switching on multithreading in Geant4

I have tried to build Geant4 with GEANT4_BUILD_MULTITHREADED flag being raised, but, during run of simulations, Allpix-Squared threw the following exception:

```
(ERROR) Library could not be loaded: not enough thread local storage available.
Try one of below workarounds:
- Rerun library with the environmental variable
LD_PRELOAD='/usr/local/Geant/lib/libG4particles.so'
- Recompile the library /usr/local/Geant4/lib/libG4particles.so with tls-
model=global-dynamic
```

The first offered solution did not help, but recompilation of the library with `tls-model=global-dynamic did helped`. I wonder if it is important taking into account the subject of the GSoC project/idea "Implement Event based Seeding and Multi-Threading".