CSEE W4840 Final Project Proposal

1. **Group**: Lotta-Pain  
   1. Andrea Lottarini al3125  
   2. Timothy K. Paine tkp2108

2. **Project**: Database Processing Unit (DPU)

3. **Introduction**: We plan to implement a custom accelerator for data analytics. This device will be similar to a streaming processor although composed by many individual components (tiles). Each of these tiles will roughly corresponds to a relational algebra operator; by composing many of them, we will be able to execute SQL queries using our device.

4. **Design Overview**: We are going to leverage some hw and sw that we already have. More specifically, we are going to translate in system verilog (and most likely expand) a set of tiles that we already have coded in verilog. Similarly we are going to utilize a compiler that we have produced capable of translating SQL queries into query plans amenable to be executed on a DPU, i.e. decomposed into a set of primitive operations that match our existing hardware. Below are our objectives for this project in detail.

5. **Hardware**:
   - We are going to translate our verilog designs of the tiles into system verilog.  
   - We are going to test them on the FPGA board and ensure that they can communicate correctly.  
   - We will probably expand and modify our design in order to allow the offload to the accelerator of a good portion of a generic SQL query.  
   - We might try to evaluate different solution for the inter-tile communication.

6. **Software**:
   - We are going to produce user level code to be run on the ARM core in order to direct the execution of the query in the FPGA board. This will be similar to a JIT or interpreter. Notice that this is different from the compiler previously mentioned as for a single query plan multiple scheduling and bindings solutions exists. These have to be evaluated at runtime when the size of the data involved is known.

7. **HW/SW integration**:
   - We are going to produce drivers and all the code necessary to have the ARM processor communicate with the tiles and the tiles to communicate between themselves.  
   - We might want to ensure that the offload mechanism is efficient and does not constitute a bottleneck for our system.

8. **Milestones**:
   - End of March: Successfully translated all hw tiles to system verilog. Tested simple interconnection between hardware tiles. Successfully read and process streams of data from the onboard memory.  
   - End of April: Have a rough version of the JIT mentioned in 6. working, i.e. capable of execute a query utilizing both the ARM core and the FPGA fabric. Drivers written for the systems
• May 15: Presentation and Demo ready. Evaluation of alternative designs for both the hw and the sw part of the project.