Optimization process:

We first tried to answer the queries normally. Splitting the queries into sub-problems, and answering those first.
Afterwards we combined these into full queries which would give the correct answer.
Then we tried to optimise these queries, look at their runtimes, see why they are slow, or too slow. We prioritized queries that had to be executed multiple times, as slow run-times here would be too costly. Whilst doing this we also looked at sub-problems that were needed for multiple queries, to later on convert into views, such as number of passed courses, or active students.
If we weren't satisfied with the query optimizations, we resorted to using materialized views, to speed up the slow part of the query. Or had to leave out the query due to time constraints.
In later stages of the assignment when we found that materialized views could be expensive so we also tried indices, where we thought they would be useful. By looking at what the specific query asked for. We also started using joins in more advanced ways, and more often which also led to some performance increases.

Chosen optimizations:

We decided to use multiple materialized views to precompute certain values, like GPA, and accrued ECTS of a StudentRegistrationId. These were named SumECTS, and StudentGPA. Using SumECTS we are able to quickly create a small view which lists studentRegistrationIds that have completed a degree, CompletedDegree. To specifically speed up Q5 we added a StudentCountPerCourse view which pre computed the number of students for a courseId. Lastly we added an index on StudentRegistrationId, Grade which significantly increased the Q1 query. Q1 went from a 10 second execution time to only 0.6 seconds. Q5 saw a reduction of 90 seconds in total execution time (5 executions) from 150 to 60 seconds. Due to the student count view, Q3 also sees a slight decrease as it can make use of the CompletedDegree view. We hoped to increase the performance of Q2 as well, as it can make use of the StudentGPA view, and the CompletedDegree view. On our laptop it was able to execute in ~40 seconds. Sadly we were not able to get the correct answer.
The space requirements of the views are relatively small compared to the tables itself. The SumECTS and the StudentGPA views are considerably smaller, only ~245MB each. The idx_grade is about ~803MB. Lastly CompletedDegree is very small consisting of only ~6472KB. So we are well within the space requirement of 4GB.