Generic Cache Module:

Problem statement:

You are required to implement a cache module/library which you will embed in your application to improve the application performance, by holding heavily accessed (read/written) application-specific objects.

Functional Requirements:

1. Your cache module should be generic, re-usable, and easy to integrate across various modules within your code/organization.
2. The cache will be bounded by a fixed capacity for holding the objects, which will be mentioned during the early initialization of the program.
3. Upon hitting the capacity, the cache module can invoke one of various cache eviction strategies to make room for newer objects.
4. You are required to incorporate cache eviction in your code to handle the aforementioned conditions.
5. You could choose to implement one or more of the varied cache eviction strategies such as 'Least recently used', 'Least frequently used', 'time-based expiration' et.al
6. Use string keys for simplicity.

Non-functional requirements:

1. We are looking for production-grade implementation with a judicious mix of code modularity, extensibility, and test coverage.
2. Usage of 3rd party libraries is not permitted.
3. Needless to say, we highly appreciate a modular, extensible implementation supported with good quality test coverage.
4. **Use git commits and include the .git directory to track the solution's progression**

Please don’t spend more than 2 hours on the solution.