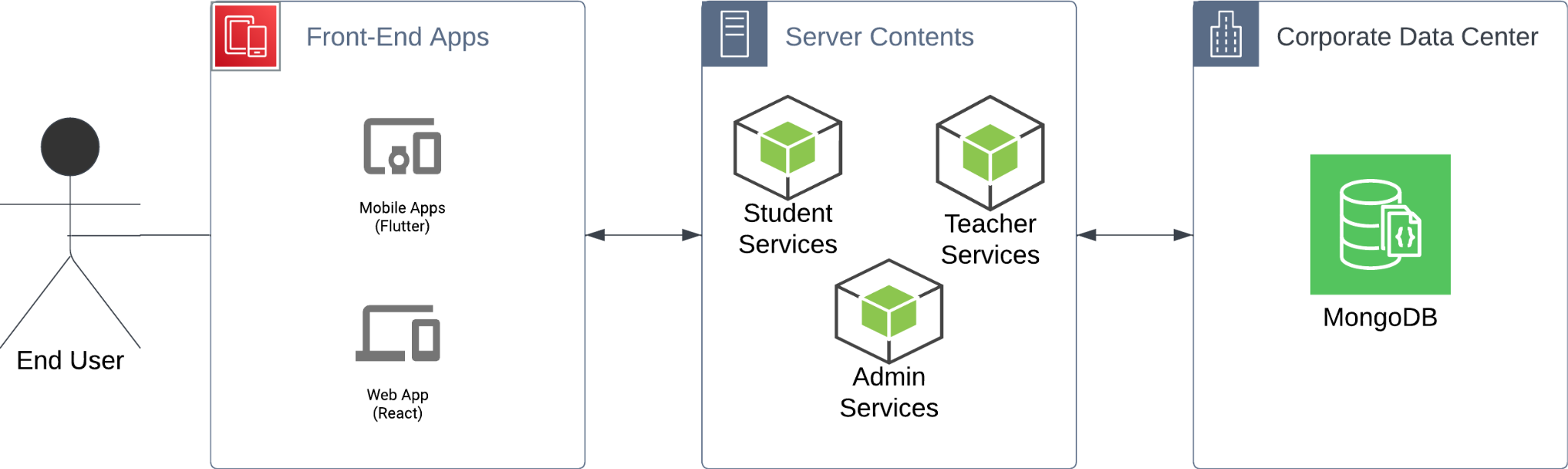
# Architecture



For Academia, the System will be split Into a simple MVC architecture with some minimalistic microservices architecture. The three main modules are independent of each other and are based on XAPI guidelines.

* FrontEnd Applications would have their own hosting server that is independent of the other modules and both simply serve as the interface to receive the data from and manage data that may be on the client side (cookies, downloadable data, uploadable files)
* On the Server The technology utilised will be Node.js Express servers as our objects are JSON-based and databases are document-based, Node.js will provide more functionality and speed to manage our data efficiently. Each stakeholder will have access or their own service which will help separate important computing resources. A teacher would not use their services as much as a student so their services can receive less computing power, while an admin analytical services may cause a big load on the servers.
* The DataBase technology will be MongoDB-based as a NoSQL database will be more suitable for the specification by the XAPI guidelines and allows for easier scalability.

Given the requirements of Academia and the stated above architecture, Special attention should be given to the database schema to save different kinds of files and to specific features like language compatibility and client-side management of files.