

Distributed Systems API

Similar to a library an API (Application Program Interface) allows consumers to call functions of code they have not necessarily written, an API differs however as unlike a library the code is not stored with the program it is instead accessed over a network typically with a HTTP request. This allows complex algorithms to have their implementations kept private but allow external parties to use them only needing to give the input and receive the output. A stateless server can be restarted and work the same as the instance that was running before it, this is because it does not store any values in memory and if storage is needed would use persistence storage. This API is stateless as nothing is stored in memory when the API is running, only the database is used to store data. A stateful server such as a game server on the other hand will use internal memory to keep track of processes for a game server this could be things like player objects and round timers.

Briefly explain what route mapping is, how WebAPI uses the id parameter and what actions are.

Briefly outline what GET, POST and DELETE requests are and provide screenshots of where you have used these requests in your server project to illustrate your written work.

Briefly describe how your Server and Client use the API key. Identify if you think an API key is a good or bad option for identifying users, giving your reasons. Is the API key safe in this project? How would you ensure this API key was kept safe if you were developing this Server/Client in the 'real world'?

Outline the steps in the RSA algorithm.

Outline the steps in the AES algorithm.

Entity Framework is an ORM (object relation manager), it allows the use of objects to interact with a database instead of traditional SQL. There are a few ways to use Entity Framework, code first allows you to write the objects that you want stored and create the database based on the models. Database first allows you to create a database and then generate the models based on the tables in the database, model first allows you to design the database with diagrams and generate the database from them you can then generate the objects you want to store the same as you would with database first.

On reflection, reflection should not have been used for writing the client, originally the calls in the client seemed to align with the Controller and Action however there were a few that didn't and the extra work to make each outlier work was not worth the time saved by using reflection.