# Customer Features Database design

* Tables

A screenshot of a cell phone

Description automatically generated

FeatureTable: Holds list of features and attributes.

CustomerFeature: Holds customer’s feature enrollment and settings.

Tools and technologies:

1. .Net Core - Microservice API are build using this.
2. MS SQL Sever – Customer Features store.
3. Dockers – Both microsevice and Sql Server are containerized.
4. Azure – An alternate platform to host database.

# Components and deployment view

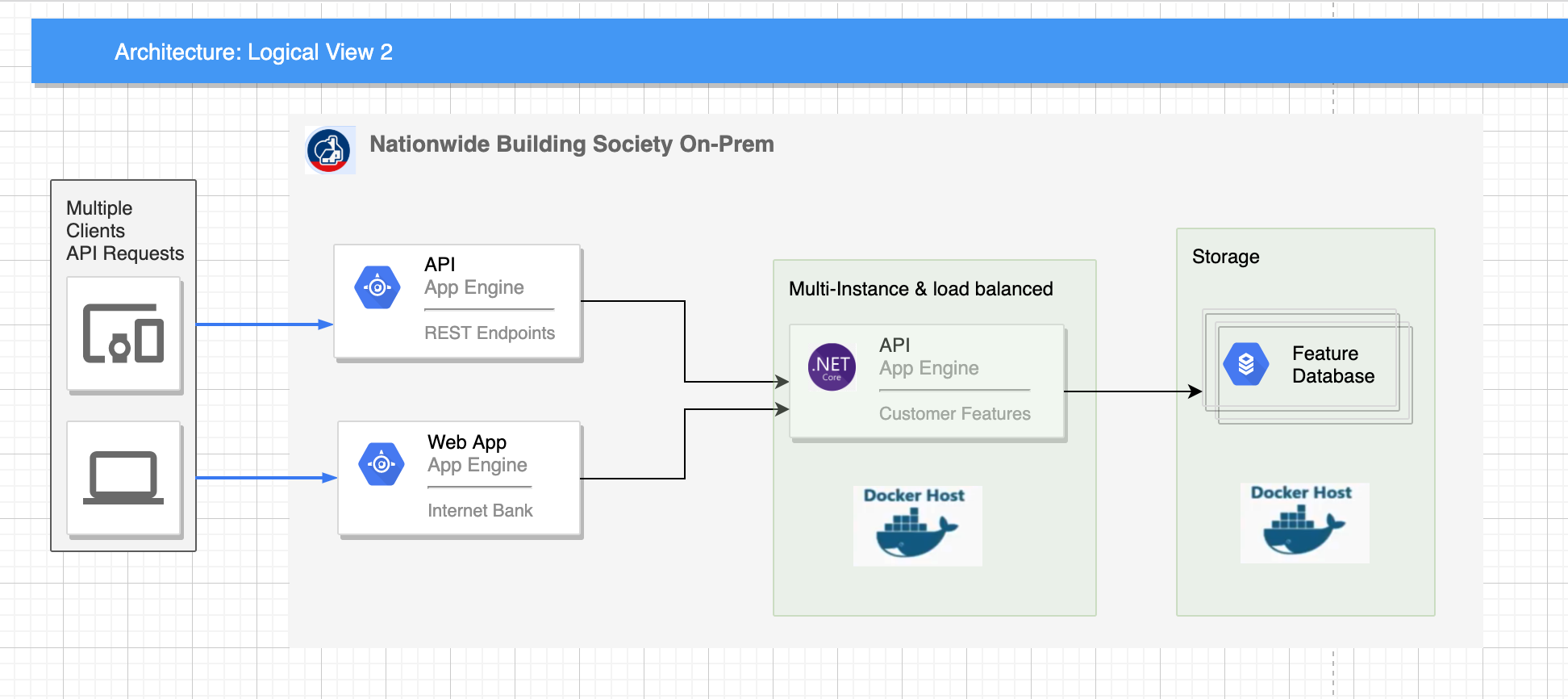
A screenshot of a cell phone

Description automatically generated

**Intent:**

Develop and host microservice built using .Net Core 3.0 on existing on-prem infrastructure of Nationwide.

.Net Core API can be hosted on existing Win 2K12 servers on IIS.



\* There is a variation, where only API was containerized, and SQL Server on existing non-containerized hosting. This has also been tried.

**Intent:**

Develop and host microservice built using .Net Core 3.0 on existing on-prem infrastructure of Nationwide using container technology.

* .Net Core containerized API can be hosted on existing Win or Linux servers.
* Containerized SQL Server database can be hosted on Win or Linux compute boxes.
* This enables us to re-platform applications on cloud platform with least effort.
* Containerization removes deployment issued due to inconsistency in development, test and production like environments.

A screenshot of a cell phone

Description automatically generated

**Intent:**

Develop and host microservice built using .Net Core 3.0 on existing on-prem infrastructure of Nationwide using container technology.

* .Net Core containerized API can be hosted on existing Win or Linux servers.
* Used Cloud native server - SQL Server database hosted on Azure as SaaS.

Back-Up

A screenshot of a social media post

Description automatically generated