Single-Tier Applications

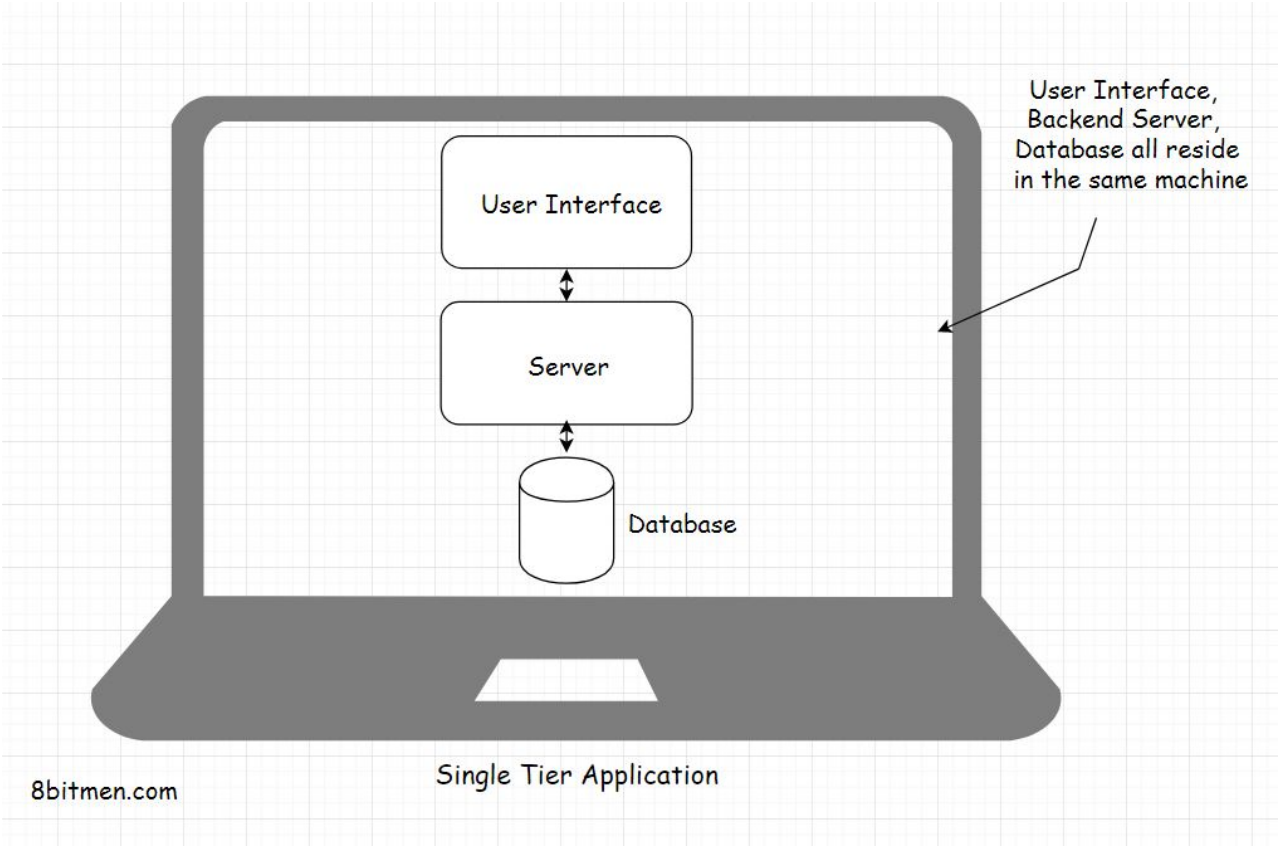
In this lesson, you will learn about single-tier applications.

We'll cover the following

- Single-tier applications
- Advantages of single-tier applications
- Disadvantages of single-tier applications

Single-tier applications#

> A *single-tier* application is an application where the *user interface*, *backend business logic*, and the *database* all reside in the same machine.



Typical examples of *single-tier* applications are desktop applications like *MS Office*, PC Games or an image editing software like *Gimp*.

Advantages of single-tier applications#

The main upside of *single-tier* applications is that they have no network latency because every component is located on the same machine. This adds up to the performance of the software.

Every now and then, data requests to the backend server stop, making the user experience slow. In *single-tier* apps, the data is easily and quickly available since it is located in the same machine.

However, the real performance of a single-tier app largely depends on how powerful the machine is and the software's hardware requirements.

Also, the user's data stays in their machine and doesn't need to be transmitted over a network. This ensures data safety at the highest level.

Disadvantages of single-tier applications#

One big downside of *single-tier* apps is that the business has no control over the application. Once the software is shipped, no code or feature changes can be done until the customer manually updates it by connecting to the remote server or by downloading and installing a patch.

Due to this, in the 90s there was nothing the studios could do if a game was shipped with buggy code. They would eventually have to face some heat due to the buggy nature of the software. The product testing had to be thorough because there was no room for any mistakes.

The code in *single-tier* applications is also vulnerable to being tweaked and reversed engineered. The security, for the business, is minimal.

Finally, the applications' performance and the look and feel can become inconsistent as the app largely depends on the configuration of the user's machine.

← Back

Introduction

Next →

Two-Tier Applications

Completed

Report an Issue