Introduction

This lesson gives an overview of the different topics we will cover in this chapter. You will learn what a tier is and its components?

We'll cover the following

- What is a tier?

I’ll begin the course by discussing the different tiers involved in the software architectural landscape. This is like a bird’s eye view of the software architecture realm and it is important to be understood well.

This chapter will help us understand:

- What is a *tier*?
- Why do software applications have different *tiers*? What is the need for them?
- How do I decide how many *tiers* my application should have?

What is a tier?#

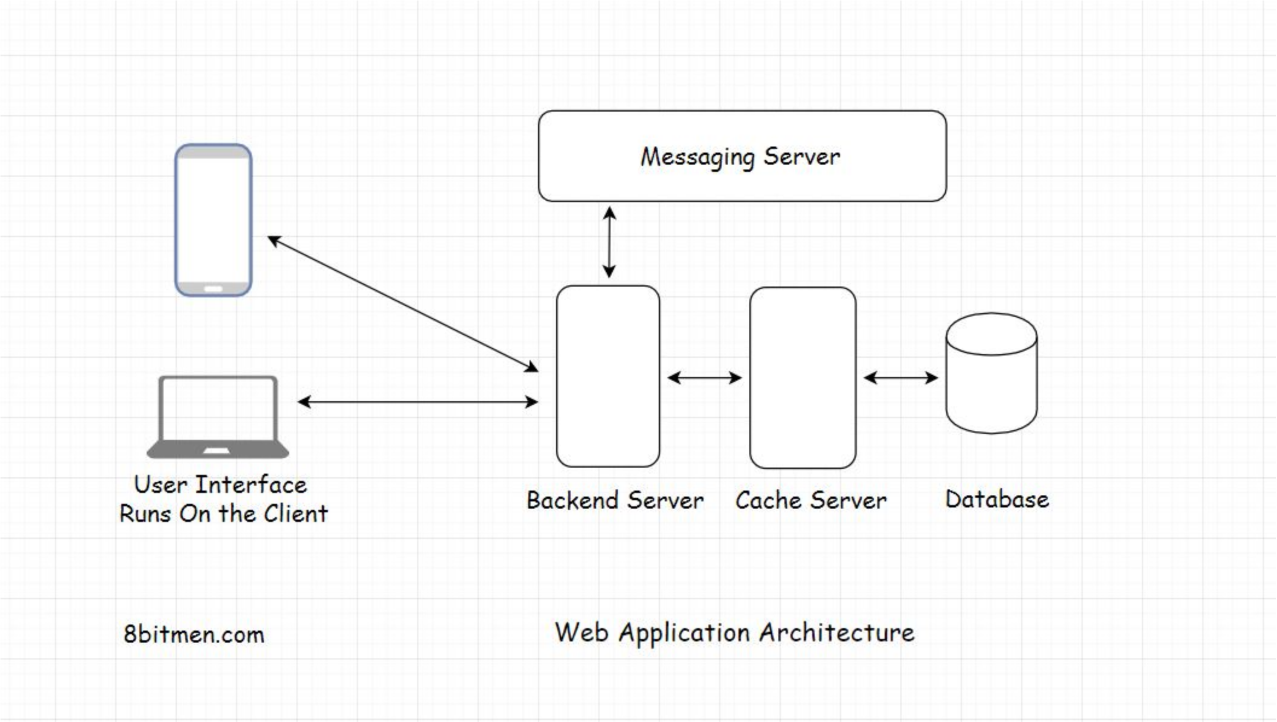
Think of a *tier* as a logical separation of components in an application or a service. When I say separation, I mean physical separation at the component level, not the code level.

What do I mean by components?

- *Database*
- *Backend application*

- *Backend application server*
- *User interface*
- *Messaging*
- *Caching*

These are the different components that make up a *web service*.



Now let's take a look at the different types of tiers and their real-life examples.

Next →

Single-Tier Applications

☑ Completed

Report an Issue

⚙ 📋