

# Conclusion

This concludes our brief look at building a simple, but fully functional, Zend Framework zend-mvc application.

In this tutorial we but briefly touched quite a number of different parts of the framework.

The most important part of applications built with zend-mvc are the modules (<https://zendframework.github.io/zend-modulemanager/intro/>), the building blocks of any zend-mvc application (<https://zendframework.github.io/zend-mvc/quick-start/>).

To ease the work with dependencies inside our applications, we use the service manager (<https://zendframework.github.io/zend-servicemanager/intro/>).

To be able to map a request to controllers and their actions, we use routes (<https://zendframework.github.io/zend-router/routing/>).

Data persistence was performed using zend-db (<https://zendframework.github.io/zend-db/adapter/>) to communicate with a relational database. Input data is filtered and validated with input filters (<https://zendframework.github.io/zend-inputfilter/intro/>), and, together with zend-form (<https://zendframework.github.io/zend-form/intro/>), they provide a strong bridge between the domain model and the view layer.

zend-view (<https://zendframework.github.io/zend-view/quick->

start/) is responsible for the View in the MVC stack, together with a vast amount of view helpers (<https://zendframework.github.io/zend-view/helpers/intro/>).

---

---

Copyright (c) 2016 Zend Technologies USA Inc. (<http://www.zend.com/>)

Learn more about Zend Framework (<http://framework.zend.com>)