The QuoteServices App Case Study: Zippy Microservice Structure & Functionality

Douglas C. Schmidt
d.schmidt@vanderbilt.edu
www.dre.vanderbilt.edu/~schmidt

Professor of Computer Science
Institute for Software Integrated Systems
Vanderbilt University
Nashville, Tennessee, USA
Learning Objectives in this Part of the Lesson

- Understand the structure & functionality of the ZippyController/ZippyService microservice & how it applies the Java Persistence Architecture (JPA)

This microservice also uses Java parallel & sequential streams
Structure & Functionality of the ZippyApplication
Structure & Functionality of the ZippyApplication

• Provides the entry point into the Spring WebMVC-based version of the Zippy Quote microservice

```java
@RestController
public class ZippyApplication
    extends BaseApplication {

    public static void main(String[] args) {
        run(ZippyApplication.class, args);
    }
}
```

See quoteservices/microservices/handey/ZippyController.java
Structure & Functionality of the ZippyApplication

- Provides the entry point into the Spring WebMVC-based version of the Zippy Quote microservice

```java
@RestController
public class ZippyApplication
    extends BaseApplication {

    public static void main(String[] args) {
        run(ZippyApplication.class, args);
    }
}
```

BaseApplication defines the run() method used by both Handey Application & ZippyApplication

See microservices/src/main/java/edu/vandy/quoteservices/common/BaseApplication.java
Structure & Functionality of the ZippyApplication

• Provides the entry point into the Spring WebMVC-based version of the Zippy Quote microservice

```java
@RestController
public class ZippyApplication
    extends BaseApplication {

    public static void main(String[] args) {
        run(ZippyApplication.class, args);
    }
}
```

Main entry-point method calls the BaseApplication helper method to build & run the microservice
Structure & Functionality
of the ZippyController
Structure & Functionality of the ZippyController

• Client HTTP GET requests are mapped to endpoint handler methods via the ZippyController subclass

```java
@RestController
public class ZippyController
    extends BaseController
    <List<Quote>> {
}
```

See quoteservices/microservices/handey/ZippyController.java
Structure & Functionality of the ZippyController

- Client HTTP GET requests are mapped to endpoint handler methods via the ZippyController subclass

```java
@RestController
public class ZippyController extends BaseController {
    <List<Quote>> {
        // Code
    }
}
```

`BaseController` defines the endpoint handler methods used by both HandeyController & ZippyController.

See `microservices/src/main/java/edu/vandy/quoteservices/common/BaseController.java`
Structure & Functionality of the ZippyController

- Client HTTP GET requests are mapped to endpoint handler methods via the ZippyController subclass

```java
@RestController
public class ZippyController extends BaseController {
    <List<Quote>>
}
```

Defines the classic Java type returned by endpoint handler methods in the ZippyController
Structure & Functionality of the ZippyController

- Client HTTP GET requests are mapped to endpoint handler methods via the ZippyController subclass

```java
@RestController
public class ZippyController
    extends BaseController
        <List<Quote>> {
}
```

This annotation ensures request handling methods in the controller class automatically serialize return objects into HttpResponse objects

See www.baeldung.com/spring-controller-vs-restcontroller
Structure & Functionality of the QuoteRepository
Structure & Functionality of the QuoteRepository

- QuoteRepository is a JPA-based persistent repository that contains information about Quote objects

```java
@Service
public class QuoteRepository
    extends JPARepository<Quote, Integer> {

    List<Quote> findByQuoteContainingIgnoreCase(String query);
}
```

See microservices/src/main/java/edu/vandy/quoteservices/common/QuoteRepository.java
Structure & Functionality of the QuoteRepository

- QuoteRepository is a JPA-based persistent repository that contains information about Quote objects

```java
@Service
public class QuoteRepository
    extends JPARepository<Quote, Integer> {

    List<Quote> findByQuoteContainingIgnoreCase(String query);
}
```

Provides methods for basic CRUD operations (e.g., save(), findAll(), findById(), findAllById(), etc.) that enable programs to interact with a database without writing any SQL code.

See [springframework/data/jpa/repository/JpaRepository.html](http://springframework/data/jpa/repository/JpaRepository.html)
Structure & Functionality of the QuoteRepository

• QuoteRepository is a JPA-based persistent repository that contains information about Quote objects

@Service
public class QuoteRepository
    extends JPARepository<Quote, Integer> {

    List<Quote> findByQuoteContainingIgnoreCase(String query);
}

Find all Quote rows in the database that contain the query String (ignoring case)

This method is automatically generated!
Structure & Functionality of the ZippyService
Structure & Functionality of the ZippyService

- ZippyService defines implementation methods called by ZippyController

```java
@Service
public class ZippyService implements BaseService<List<Quote>> {
    @Autowired
    private QuoteRepository mRepository;
    ...
```
Structure & Functionality of the ZippyService

• ZippyService defines implementation methods called by ZippyController

@Service
public class ZippyService implements BaseService<List<Quote>> {
    @Autowired private QuoteRepository mRepository;
    ...

See microservices/src/main/java/edu/vandy/quoteservices/common/BaseService.java

The BaseService defines the methods overridden by both HandeyService & ZippyService
• ZippyService defines implementation methods called by ZippyController

@Service
public class ZippyService implements BaseService<List<Quote>> {
    @Autowired private QuoteRepository mRepository;
    ...

Defines the classic Java type returned by endpoint handler methods in the ZippyService
Structure & Functionality of the ZippyService

- ZippyService defines implementation methods called by ZippyController

```java
@Service
public class ZippyService implements BaseService<List<Quote>> {
    @Autowired private QuoteRepository mRepository;
    ...
}
```

This annotation indicates the class implements "business logic" & enables auto-detection & wiring of dependent classes via classpath scanning

See [www.baeldung.com/spring-component-repository-service](http://www.baeldung.com/spring-component-repository-service)
Structure & Functionality of the ZippyService

- ZippyService defines implementation methods called by ZippyController

```java
@Service
public class ZippyService implements BaseService<List<Quote>> {
    @Autowired private QuoteRepository mRepository;
    ...
```

See [www.baeldung.com/spring-autowire](http://www.baeldung.com/spring-autowire)
Structure & Functionality of the ZippyService

- ZippyService defines implementation methods called by ZippyController

```java
@Service
public class ZippyService implements BaseService<List<Quote>> {
    public List<Quote> getAllQuotes() {
        return mRepository.findAll();
    }
    ...
}
```

*This method uses the JPA to return all the Zippy quotes*
Structure & Functionality of the ZippyService

- ZippyService defines implementation methods called by ZippyController

```java
@Service
public class ZippyService implements BaseService<List<Quote>> {
    public List<Quote> postQuotes(List<Integer> quoteIds, Boolean parallel) {
        return mRepository.findAllById(quoteIds)
    }
}
```

This method uses the JPA to find all quotes matching the quoteIds.
Structure & Functionality of the ZippyService

- ZippyService defines implementation methods called by ZippyController

```java
@Service
public class ZippyService implements BaseService<List<Quote>> {
    public List<Quote> search(List<String> queries, Boolean parallel) {
        return StreamSupport.stream(queries.spliterator(), parallel)
            .flatMap(query -> mRepository.findByQuoteContainingIgnoreCase(query).stream())
            .toList();
    }
    // ...
End of the QuoteServices App Case Study: Zippy MicroService Structure & Functionality