

The QuoteServices App Case Study: Zippy Microservice Structure & Functionality (Part 1)

Douglas C. Schmidt

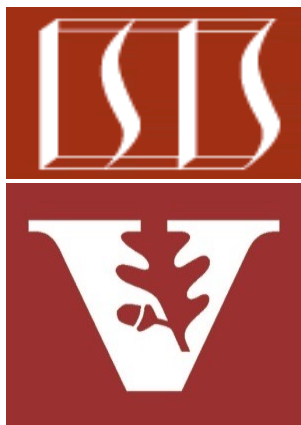
d.schmidt@vanderbilt.edu

www.dre.vanderbilt.edu/~schmidt

Professor of Computer Science

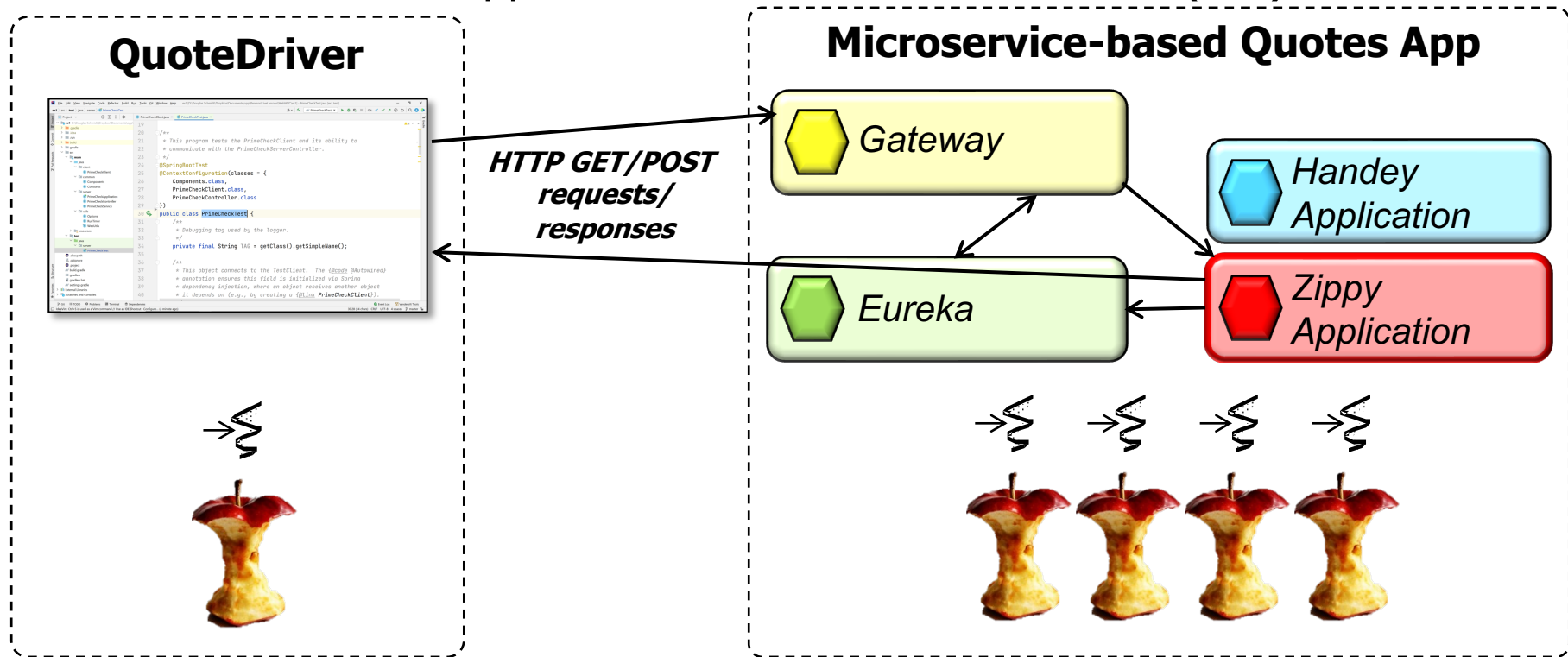
**Institute for Software
Integrated Systems**

**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 Jakarta Persistence API (JPA)



This microservice also uses Java parallel & sequential streams a bit..

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

```
@SpringBootApplication
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

```
@SpringBootApplication  
public class ZippyApplication  
    extends BaseApplication {
```

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

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

Structure & Functionality of the ZippyApplication

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

```
@ SpringBootApplication
public class ZippyApplication
    extends BaseApplication {
```

*Main entry-point method calls the
BaseApplication helper method
to build & run the microservice*

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

Structure & Functionality of the ZippyController

Structure & Functionality of the ZippyController

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

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

See [quoteservices/microservices/handey/ZippyController.java](https://github.com/quoteservices/microservices/handey/ZippyController.java)

Structure & Functionality of the ZippyController

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

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

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

Structure & Functionality of the ZippyController

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

```
@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

```
@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

Structure & Functionality of the QuoteRepository

Structure & Functionality of the QuoteRepository

- JPAQuoteRepository is a Jakarta Persistence API (JPA)-based repository that contains a database of Quote objects

`@Service`

```
public class JPAQuoteRepository  
    extends JpaRepository<Quote, Integer>,  
        MultiQueryRepository {
```



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

Structure & Functionality of the QuoteRepository

- JPAQuoteRepository is a Jakarta Persistence API (JPA)-based repository that contains a database of Quote objects

```
@Service
```

```
public class JPAQuoteRepository  
    extends JpaRepository<Quote, Integer>,  
           MultiQueryRepository {
```

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

See [quoteservices/microservices/zippy/JPAQuoteRepository.java](https://github.com/zippp/JPAQuoteRepository.java)

Structure & Functionality of the QuoteRepository

- JPAQuoteRepository is a Jakarta Persistence API (JPA)-based repository that contains a database of Quote objects

```
@Service
public class JPAQuoteRepository
    extends JpaRepository<Quote, Integer>,
        MultiQueryRepository {
```

Provides methods for basic CRUD operations (e.g., save(), findAll(), findById(), findAllById(), etc.) that enable interactions with a database while writing no SQL

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

See [springframework/data/jpa/repository/JpaRepository.html](https://springframework.org/data/jpa/repository/JpaRepository.html)

Structure & Functionality of the QuoteRepository

- JPAQuoteRepository is a Jakarta Persistence API (JPA)-based repository that contains a database of Quote objects

@Service

```
public class JPAQuoteRepository
    extends JpaRepository<Quote, Integer>,
        MultiQueryRepository {
```

*Provides a method for returning a List of Quote objects
that match all queries in a List of String objects*

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

See [quoteservices/microservices/zippy/MultiQueryRepository.java](https://github.com/zipper-io/zipper/blob/master/src/main/java/com/zipperio/quoteservices/microservices/zippy/MultiQueryRepository.java)

Structure & Functionality of the QuoteRepository

- JPAQuoteRepository is a Jakarta Persistence API (JPA)-based repository that contains a database of Quote objects

@Service

```
public class JPAQuoteRepository  
    extends JpaRepository<Quote, Integer>,  
        MultiQueryRepository
```

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

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

Structure & Functionality of the QuoteRepository

- JPAQuoteRepository is a Jakarta Persistence API (JPA)-based repository that contains a database of Quote objects

@Service

```
public class JPAQuoteRepository  
    extends JpaRepository<Quote, Integer>,  
        MultiQueryRepository
```

This JPA Data API method implementation is automatically generated as
`SELECT * FROM quote WHERE LOWER(quote) LIKE LOWER('%{query}%')`

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

See part 2 of this lessons on "Zippy Microservice Structure & Functionality"

Structure & Functionality of the ZippyService

Structure & Functionality of the ZippyService

- ZippyService defines implementation methods called by ZippyController

`@Service`

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

Structure & Functionality of the ZippyService

- ZippyService defines implementation methods called by ZippyController

`@Service`

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

The BaseService defines the methods overridden by both HandeyService & ZippyService

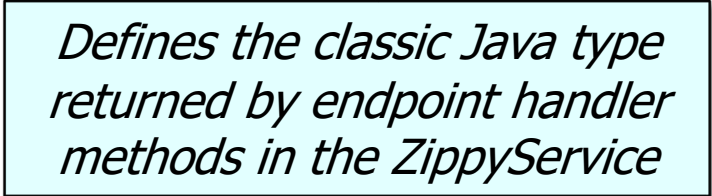
See [microservices/src/main/java/edu/vandy/quoteservices/common/BaseService.java](https://github.com/microservices/src/main/java/edu/vandy/quoteservices/common/BaseService.java)

Structure & Functionality of the ZippyService

- ZippyService defines implementation methods called by ZippyController

`@Service`

```
public class ZippyService implements BaseService<List<Quote>> {  
    @Autowired private JPAQuoteRepository 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

@Service

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

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

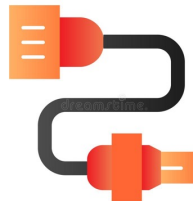
See www.baeldung.com/spring-component-repository-service

Structure & Functionality of the ZippyService

- ZippyService defines implementation methods called by ZippyController

`@Service`

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



*This field is auto-wired
by Spring's dependency
injection framework*

Structure & Functionality of the ZippyService

- ZippyService defines implementation methods called by ZippyController

@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*

The generated SQL query would be "SELECT * FROM quote"

Structure & Functionality of the ZippyService

- ZippyService defines implementation methods called by ZippyController

@Service

```
public class ZippyService implements BaseService<List<Quote>> {  
    ...  
    public List<Quote> postQuotes(List<Integer> quoteIds,  
                                   Boolean notUsed) {  
        return mRepository  
            .findAllById(quoteIds);  
    } ...  
}
```

*This method uses the JPA to find
all quotes matching the quoteIds*

The generated SQL query would be "SELECT * FROM quote WHERE id IN (?, ?, ...)"

Structure & Functionality of the ZippyService

- ZippyService defines implementation methods called by ZippyController

@Service

```
public class ZippyService implements BaseService<List<Quote>> {  
    ...  
    public List<Quote> search(List<String> queries,  
                               Boolean parallel) {  
        return StreamSupport  
            .stream(queries.splititerator(), parallel)  
            .flatMap(query -> mRepository  
                .findByQuoteContainingIgnoreCase(query)  
                .stream())  
            .distinct()  
            .toList();  
    } ...  
}
```

Uses Java parallel & sequential streams & the JPA to find Quote objects matching any query

The SQL query is "SELECT * FROM quote WHERE LOWER(quote) LIKE LOWER('%{query}%')"

Structure & Functionality of the ZippyService

- ZippyService defines implementation methods called by ZippyController

@Service

```
public class ZippyService implements BaseService<List<Quote>> {  
    ...  
    public List<Quote> searchEx(List<String> queries,  
                                Boolean notUsed) {  
        return mRepository  
            .findAllByQuoteContainingAllIn(queries);  
    } ...  
}
```



This method uses a custom SQL query to find all Quote objects containing all the queries

See part 2 of this lessons on "Zippy Microservice Structure & Functionality"

End of the QuoteServices App Case Study: Zippy MicroService Structure & Functionality (Part 1)