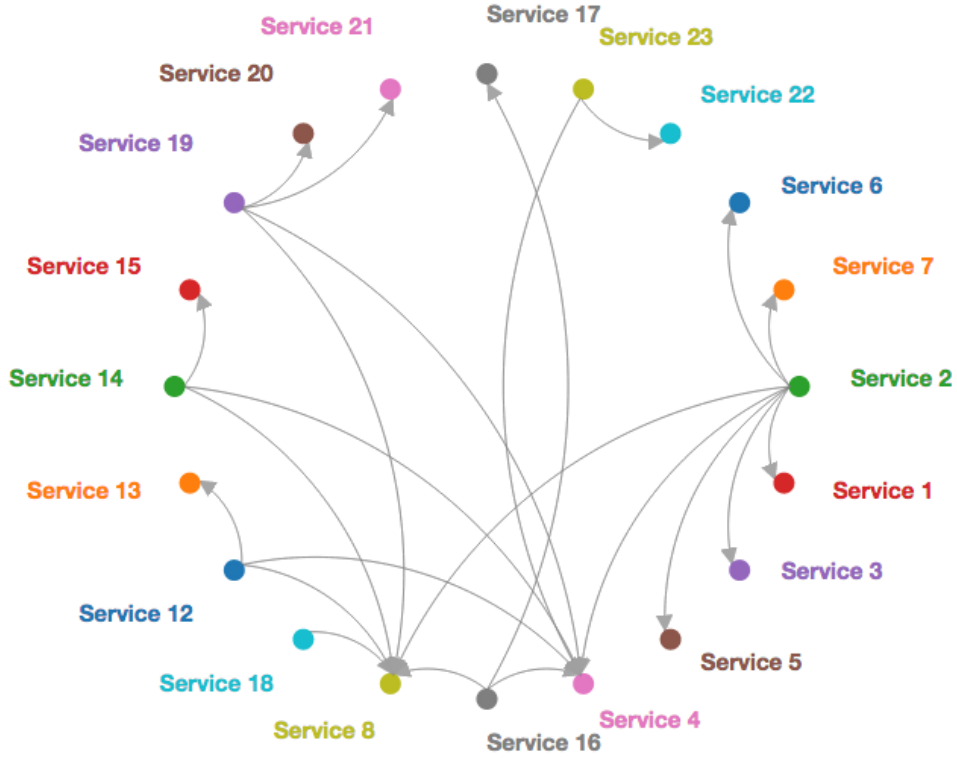


Microservice Test

Microservice Mimariler’de servis sayısı arttıkça, servisler arası iletişim ve entegrasyon sayısı da doğru orantılı olarak artacaktır. 10 adet servisten oluşan bir sistem ile 100 adet veya daha fazla servisin birbirleriyle konuştuğu bir sistemin karmaşıklığı ve bakım maliyetleri hiç şüphesiz aynı olmayacaktır.



Eğer az sayıda Microservice’e sahip bir uygulamamız varsa, söz konusu ekipte küçük bir ekiptir muhtemelen ve bunu belki bir şekilde manuel ilerletebilirsiniz. Burada manuel’den kastımız ekip içerisinde, servislerde yapılan değişikliklerin, değişiklikten etkilenen servislerin sorumlularına doğrudan bildirilmesidir. Ancak onlarca hatta yüzlerce servisten bahsediyorsak bunu manuel olarak yönetmemiz söz konusu olmaz.

Spring Test Nasıl Yapılır:

Örnek uygulama: <https://www.baeldung.com/spring-test-pyramid-practical-example>

MovieController.java

```
@RestController
public class MovieController {

    @Autowired
    private MovieService movieService;

    @GetMapping("/movies")
    public List<Movie> retrieveAllMovies() {
        return movieService.retrieveAllMovies();
    }

    @GetMapping("/movies/{id}")
    public Movie retrieveMovies(@PathVariable Long id) {
        return movieService.retrieveMovies(id);
    }

    @PostMapping("/movies")
    public Long createMovie(@RequestBody Movie movie) {
        return movieService.createMovie(movie);
    }
}
```

MovieServiceUnitTests.java

```
1 public class MovieServiceUnitTests {
2
3     @InjectMocks
4     private MovieService movieService;
5
6     @Mock
7     private MovieRepository movieRepository;
8
9     @Before
10    public void setUp() throws Exception {
11        MockitoAnnotations.initMocks(this);
12    }
13
14    @Test
15    public void givenMovieServiceWhenQueriedWithAnIdThenGetExpectedMovie() {
16        Movie movie = new Movie(100L, "Hello World!");
17        Mockito.when(movieRepository.findById(100L))
18            .thenReturn(Optional.ofNullable(movie));
19
20        Movie result = movieService.retrieveMovies(100L);
21
22        Assert.assertEquals(movie.getTitle().toLowerCase(), result.getTitle());
23    }
24 }
```