# **Uppgift: JS-Ramverk**

# **Namn: Viktor Ek**

**Utförande**

Först och främst vill jag att du skriver generellt om ramverk och varför de finns? Vilka är fördelarna? Finns det några nackdelar? Klarar man sig utan ett ramverk eller måste man ha ett om man ska utveckla en större sida?

*Skriv här…* Ramverk är omfattande plugins eller hjälpbibliotek, som används för att förbättra redan existerande programmeringsspråk. Ramverk används för att lägga till funktionalitet eller för att förenkla hantering och processer. De kan ses som verktyg eller tillägg till ett programmeringsspråk för att göra det enklare för programmerare att utföra vissa operationer.

Man ska inte använda ramverk om det inte behövs för att koden blir längre vilket betyder att webbsidan blir “tyngre” I form av GB vilket gör att den får sämre “performance”. Det finns ramverk som är väldigt komplicerade och kan bli svårare att hantera men sen finns det ju även de som är väldigt bra och hjälper väldigt mycket.

Man kan klara sig utan ett ramverk när man utvecklar en större sida men det hjälper väldigt mycket på vägen och gör det enklare för utvecklaren.

Sedan vill jag att du researchar kring olika ramverk som finns bland javascript och listar upp tre (3) populära ramverk med var sin rubrik.

Under varje ramverk ska du skriva och förklara, så utförligt du kan om följande:

Angular ***#1***

● Hur implementeras ramverket? Finns det olika sätt att implementera?

*Skriv… Den implementeras för att utveckla en “Single Page Application”*

● Hur ser syntaxen ut i respektive ramverk? Infoga gärna en kodsnutt.

*Skriv…* <h2>Products</h2>

<div \*[ngFor](https://angular.io/api/common/NgForOf)="let product of products">

<h3> <a [title]="product.name + ' details'"> {{ product.name }} </a> </h3>

<p \*[ngIf](https://angular.io/api/common/NgIf)="product.description"> Description: {{ product.description }} </p>

<button (click)="share()"> Share </button>

</div>

import { [Component](https://angular.io/api/core/Component), [OnInit](https://angular.io/api/core/OnInit) } from '@angular/core'; @[Component](https://angular.io/api/core/Component)({ selector: 'app-product-alerts', [templateUrl](https://angular.io/api/core/Component#templateUrl): './product-alerts.component.html', [styleUrls](https://angular.io/api/core/Component#styleUrls): ['./product-alerts.component.css'] })

export class ProductAlertsComponent implements [OnInit](https://angular.io/api/core/OnInit) { constructor() { } ngOnInit() { } }

● När ska man använda sig av detta ramverk och varför är det så speciellt?

*Skriv… Angular används speciellt åt “Single page applications” vilket tillexempel är sidor som har likes, kommentarer, och allt i den stilen. Den är så speciell för att* Angular hanterar dina applikationskomponenter och fungerar som en kanal som integrerar dessa komponenter till en enda applikation igen. Det låter utvecklare skapa specialiserade appar på kortare tid än andra metoder.

***Vue #2***

● Hur implementeras ramverket? Finns det olika sätt att implementera?

*Skriv… Vue komponenterna implementeras via det speciala attributet “is”, och den stöds av safari 10.1+ chrome 54+ och firefox 63+ men inte av alla.*

● Hur ser syntaxen ut i respektive ramverk? Infoga gärna en kodsnutt.

*Skriv…*

|  |
| --- |
| <template> |
|  | <div> |
|  | {{ foo \* 10 + 'hi' }} |
|  | <span |
|  | v-text="foo \* 10 + 'hi'" |
|  | :id="foo + 'baz'" |
|  | @click="onClick('hello')" |
|  | @click.prevent="onClick('hello')"> |
|  | Hello |
|  | </span> |
|  | </div> |
|  | </template> |

● När ska man använda sig av detta ramverk och varför är det så speciellt?

*Skriv…Vue funkar både på “Single page applications” och “Multi page applications” så du kan använda det på vilket som, men det är så speciellt för att det är enklare att lära sig än både Angular och React och det är de lättaste att vänja sig vid för det använder redan problemlösningar som vi är vana vid.*

***React #3***

● Hur implementeras ramverket? Finns det olika sätt att implementera?

*Skriv… Via en massa kod och det finns olika kodsnuttar om implementerar React.*

<https://reactjs.org/docs/implementation-notes.html>

● Hur ser syntaxen ut i respektive ramverk? Infoga gärna en kodsnutt.

*Skriv…***import** React from 'react';

2 **import** { AppRegistry, Text } from 'react-native';

3

4 **const** HelloWorldApp = () => <Text>Hello world!</Text>;

5 **export** **default** HelloWorldApp;

6

7 *// Skip this line if using Create React Native App*

8 AppRegistry.registerComponent('HelloWorld', () => HelloWorldApp);

9

10 *// The React native code can also be imported from another component with the following code:*

11 **import** HelloWorldApp from './HelloWorldApp';

● När ska man använda sig av detta ramverk och varför är det så speciellt?

*Skriv… React är nog bäst att använda för simpla ”one page sites”. Så ska du bara framställa 50 000 saker eller mindre behöver du inte använda react, Eller om du ska göra en riktigt stor sida kan react också vara bra att använda.*

**Är det något särskilt du kommer minnas efter denna uppgift och ta med dig i din roll som utvecklare? Var det något ramverk du fastnade mer för än andra?**

*Svara här… Vad ramverken har för påverkan på koden och vad de används till. Fastnade för Vue mer än dom andra för det var det enklare av dem tre.*