|  |
| --- |
|  |
| Wearep |
|  |
| En app för att rapportera väderdata för en utvald plats på twitter eller i html. |

**SYSTEMINTEGRATÖR YHSIPI17/Datakommunikation och nätverk**

den 1 oktober 2018

Billy Andersson, Malin Albinsson, Sten Karlsson

Wearep

En app för att rapportera väderdata för en utvald plats på twitter eller i html.

Innehåll

[2 Inledning 2](#_Toc526375050)

[2.1 Sammanfattning TODO 2](#_Toc526375051)

[2.2 Inledning/Syfte 2](#_Toc526375052)

[2.3 Definitioner och förkortningar TODO 2](#_Toc526375053)

[2.4 Referenser TODO 2](#_Toc526375054)

[3 Genomförande 2](#_Toc526375055)

[3.1 Övergripande mål TODO 2](#_Toc526375056)

[3.2 avgränsningar TODO 2](#_Toc526375057)

[3.3 Arbetsmetodik TODO 3](#_Toc526375058)

[4 Wearep-appen 3](#_Toc526375059)

[4.1 Översikt TODO 3](#_Toc526375060)

[4.2 Platsdata från Google Maps API TODO 3](#_Toc526375061)

[4.2.1 Google Maps API TODO 3](#_Toc526375062)

[4.3 Väderdata från SMHIs API TODO 3](#_Toc526375063)

[4.3.1 SMHIs API TODO 3](#_Toc526375064)

[4.4 Twitter-möjlighet från Twitters API TODO 3](#_Toc526375065)

[4.4.1 Twitters API TODO 3](#_Toc526375066)

[4.5 Publicera HTML 3](#_Toc526375067)

[4.6 Teknisk lösning 3](#_Toc526375068)

[4.6.1 JavaFx TODO 3](#_Toc526375069)

[4.6.2 Utvecklingsmiljö och utvecklingsverktyg TODO 3](#_Toc526375070)

[4.6.3 Felhantering 4](#_Toc526375071)

[4.6.3.1 Regler 4](#_Toc526375072)

[4.6.3.2 Dataformatering 4](#_Toc526375073)

[4.7 Säkerhet TODO 4](#_Toc526375074)

[4.7.1 Autenticiering TODO 4](#_Toc526375075)

[4.7.2 Dataexponering TODO 4](#_Toc526375076)

[4.7.3 Sessionshantering TODO 4](#_Toc526375077)

[4.7.4 Sårbarheter TODO 4](#_Toc526375078)

[5 Slutsatser/Diskussion 4](#_Toc526375079)

[5.1 Appens implementering TODO 4](#_Toc526375080)

[5.2 Säkerhetsaspekter TODO 4](#_Toc526375081)

[5.3 Resultat TODO 4](#_Toc526375082)

# Inledning

## Sammanfattning TODO

Wearep är en app för att rapportera väderdata för en utvald plats på twitter eller i html.TODO Vi valde en MIT-licens därför att detta är en enkel applikation där källkoden kommer vara åtkomlig för alla via github. Det är inte heller något nytt, komplicerat eller innovativt så det finns anledning till större begränsningar av den anledningen.

## Inledning/Syfte

Vår uppgift var att leverera ett projekt som skulle skrivas i Java och ha någon form av koppling till webb, exempelvis interaktion med Twitters eller Facebooks APIer. JSON eller XML skulle ingå i flödet för programmet, alternativt att hämta en hemsida och med hjälp av en DOM-parser hämta ut information från den, det vill säga, projektet skulle visa att vi kan hämta information från JSON, XML eller DOM-struktur i HTML.

Programmet behövde inte vara supersnyggt. Grafiskt eller kommandoradsbaserat spelade ingen roll. Det var tillåtet att använda andra komponenter som databaser eller javabibliotek för att lösa uppgiften.

Huvuduppgiften var att visa att vi kan hantera nätverksanslutningar från Java, med fokus på webben. Vi skulle också beskriva hur vi analyserat säkerhetsaspekten i det program vi skrivit, vad vi tänkt på och hur vi löst det. Källkoden skulle versionshanteras med git.

Vi valde att bygga en app som kan twittra, hämta och publicera väderdata för en utvald plats endera som html eller i en tweet.

## Definitioner och förkortningar TODO

## Referenser TODO

Länk till git-repo

Länk till Trello-tavla

Twitters API-dokumentation

SMHIs API-dokumentation

Google Maps API-dokumentation

# Genomförande

## Övergripande mål TODO

## avgränsningar TODO

Felhanteringen, testningen, kan inte välja vilken väderdata som visas

## Arbetsmetodik TODO

Vi använde ett repository på github.com TODO länk där vi alla hade skrivrättigheter. Vi förde det mesta av diskussionerna via Slack. Trello fick fungera både för KANBAN och

Licens TODO MIT?

Initialt fördelade vi arbetet så att vi arbetade med varsitt API; Billy med Twitter, Sten med Google Maps och Malin med SMHI. Därutöver arbetade vi med…TODO. Varje API fick varsin branch (även om de fick namn efter oss), JavaFX-skalet fick en egen och därutöver hade vi en test-branch för att kunna testa att de olika delarna fungerar tillsammans innan vi släppte dem till master.

# Wearep-appen

## Översikt TODO

Appen Wearep kan logga in på Twitter, hämta och posta tweets. Den kan också hämta och publicera väderdata från SMHI för en utvald plats endera som html eller i en tweet. Platsens koordinater hämtas från Google Maps. Vi har således interagerat med tre olika API:er.

## Platsdata från Google Maps API TODO

TODO Översiktlig text här

### Google Maps API TODO

## Väderdata från SMHIs API TODO

Väderdata hämtas från SMHIs API (TODO länk) i JSON-format. JSON-objektet bryts ner i Weather-klassen så att enskilda variabler kan avläsas. Klassen levererar en JSON-array som endera utnyttjas av klassen WeatherSelector där förhandsdefinierade variabler för den närmaste timmens prognos plockas ut och packas ihop i ett nytt JSON-objekt, eller används direkt i klassen HtmlPrinter.

### SMHIs API TODO

## Twitter-möjlighet från Twitters API TODO

### Twitters API TODO

## Publicera HTML

Klassen HtmlPrinter omvandlar JSON-objektet från WeatherSelector till en sträng som kan hämtas upp och twittras ut. Klassen bryter även ner JSON-arrayen från Weather till en sträng och skriver den till html.

## Teknisk lösning

### JavaFx TODO

### Utvecklingsmiljö och utvecklingsverktyg TODO

Följande miljöer och verktyg användes för realisering, byggande och tester:

|  |  |
| --- | --- |
| Produkt | Ansvar |
| Eclipse | Programmeringsverktyg (IDE) |
| JUnit | Enhetstestning av kod |
| xxx | Byggstöd och rapporteringsverktyg |

### Felhantering

Javas Exceptions används för felhantering i applikationen.

#### Regler

När ett fel inträffar ska problemet i första hand åtgärdas. Om felhanteringsrutinen inte kan åtgärda problemet ska det kastas vidare.

De fel som kastas ska i första hand vara egendefinierade, för att dölja implentationsdetaljer som inte bör framgå i det publika användargränssnittet. På så vis förbättras också systemets underhållsbarhet.

#### Dataformatering

Datum skall formateras till ÅÅÅÅ-MM-DD HH24MM format, t ex 2018-07-12 15:00.

## Säkerhet TODO

### Autenticiering TODO

TODO

### Dataexponering TODO

TODO

### Sessionshantering TODO

TODO

### Sårbarheter TODO

TODO

# Slutsatser/Diskussion

## Appens implementering TODO

TODO

## Säkerhetsaspekter TODO

TODO

## Resultat TODO

TODO