Onderzoek 2FA Service

Fontys | TILBURG

Quinn richardson, Kayle Boersen

2022

# Inleiding

Dit document behoord tot de opdracht Ontdekstation. Voor deze opdracht zullen wij een bestaande website verder uitwerken met niet alleen nieuwe functionaliteiten, maar zullen wij ook een nieuwe look geven aan de website zodat het op een manier nog gebruikersvriendelijker wordt. Dit onderzoek focust zich specifiek op de verificatie functionaliteit.

# Contents

[Inleiding 1](#_Toc756386946)

[Contents 2](#_Toc1704082137)

[Opdracht 3](#_Toc1647980034)

[Onderzoek 4](#_Toc235827008)

[2.1 SMS 5](#_Toc1340193263)

[2.1.1 Twilio 5](#_Toc1114929846)

[2.1.2 Plivio 5](#_Toc101997653)

[2.1.3 Amazon AWS 5](#_Toc1255050623)

[2.2 Email 5](#_Toc370578690)

[2.2.1 Google 6](#_Toc477751203)

[2.2.2 Sendinblue 6](#_Toc1356954523)

[2.2.3 Zelf hosten 6](#_Toc482639116)

[2.2.4 Webhosting 6](#_Toc936516199)

[2.2.1 Conclusie e-mail 6](#_Toc248430665)

[2.3 Specificaties 6](#_Toc713977272)

[2.3.1 Verificatie type 7](#_Toc1186812083)

[2.3.2 Veiligheid 7](#_Toc1601779760)

[2.3.3 2-factor authentication 7](#_Toc279235890)

[3.1 Email 8](#_Toc919839682)

[3.1.1 Eerste email fase 8](#_Toc2132650601)

[3.1.2 Uitleg bestanden 8](#_Toc699802669)

[3.1.3 MIME standaarden 10](#_Toc344144660)

[3.1.4 Samenvatting 11](#_Toc532002210)

[3.2 Verdere uitwerking 12](#_Toc658418823)

[Conclusie 13](#_Toc1709570272)

[Sources 14](#_Toc730109420)

# Opdracht

Momenteel is er in de applicatie geen manier voor gebruikers om hun contact gegevens in te vullen, en ook niet om de contact gegevens te bevestigen. In overleg met de opdrachtgever willen wij de volgende requirements toevoegen;

FR-12: De applicatie heeft de mogelijkheid om gebruikers te contacteren.

FR-12.1 Bij het invullen van contactgegevens moeten deze geverifieerd worden door middel van een verstuurd bericht.

In deze requirement moet de applicatie de functie hebben om of automatisch, of vanuit een administrator de gebruikers kunnen contacteren. Wij zijn met de opdrachtgever tot de conclusie gekomen dat onder andere email en SMS goede en veelgebruikte opties zijn. Om het email adres en/of telefoonnummer te verifiëren of het daadwerkelijk van de eigenaar is willen we de gebruiker een code of link sturen die we dan op de website controleren. Als de email of sms dan gecontroleerd is kan de gebruiker door middel hiervan gebruik gaan maken van een 2 factor authentication (2FA in het kort) systeem. Wat wij hier bedoelen met een 2FA is eigenlijk de mogelijkheid om een bericht te sturen naar het email of telefoonnummer dat de gebruiker heeft toegevoegd op de website. In dit bericht wat wij sturen zal zich een code bevinden die de gebruiker terug op de website in zou moeten vullen om te bevestigen dat dit geen nep email of telefoonnummer is.

Wij zijn samen met de klant tot de conclusie gekomen waarom wij dit willen doen. De reden is om er zeker van te zijn dat het email of telefoonnummer van de gebruiker dat in is gevuld ook daadwerkelijk klopt. Het is van belang dat Ontdekstation contact op kunnen nemen met hun gebruikers als er iets misgaat met hun stations. Ook zal het gebruik van een verificatie systeem, tot een zekere graad, nepaccounts verminderen.

In het onderzoek willen we onderzoeken:

* Welke mogelijkheden en services zijn beschikbaar voor klantcontact met E-mail en SMS?
* Het verschil tussen verschillende sms-services, en verschillende email services en mogelijkheden.
* Beste manier om verificatie te kunnen integreren.

Daarnaast willen we van een van de verificatie mogelijkheden onderzoeken of het mogelijk is om deze functie te implementeren in hetzelfde framework als het bestaande project.

# Onderzoek

## 2.1 SMS

### 2.1.1 Twilio

Het lijkt erop dat een veel gebruikte service Twilio is. Twilio maakt gebruik van web sockets om een sms te sturen. Om deze service te gebruiken moet je een telefoonnummer kopen wat via de Twilio website gedaan kan worden. Dit telefoonnummer zal 5 euro per maand kosten volgens de website prijzen. Nadat je dit gedaan hebt zal er een koste zijn per bericht dat gestuurd wordt. Het lijkt erop dat elke provider die de website aanbiedt een koste rekent van 0.0966 dollar per bericht.

Helaas is Twilio niet een service dat opensource is. Omdat de Ontdekstation eigenlijk zo open source mogelijk wil zijn is Twilio niet een goede keuzen.

2.1.2 Plivio  
In het onderzoek naar een sms-service kwam ook Plivio omhoog. De prijzen komen overheen met wat Twilio aanbiedt. Het ding met Plivio is dat Plivio meer gericht is op een twee wegens communicatie. Dit betekent dat het doeleinde van Plivo meer gericht is op de mogelijkheid voor gebruikers een sms terug kunnen sturen.

### 2.1.3 Amazon AWS

Twilio zelf maakt gebruik van Amazon ’s AWS. Hierom hebben we ook daarna gekeken. Het probleem is hier ook dat AWS niet open source is wat tegen de wens van de klant ingaat. Ondanks dit probeerde wij nog verder onderzoek erop te doen alleen kwamen we al snel tegen een muur aan dat we een creditcard moesten toevoegen om de mogelijkheden van de service te kunnen zien.

## 2.2 Email

Voor het sturen van e-mails vanuit de applicatie is het nodig om een SMTP (Simple Mail Transfer Protocol) server te hebben. Een SMTP-server is applicatie op een server om e-mails te kunnen verzenden en te ontvangen. Er zijn verschillende bedrijven die ze hosten voor klanten die allemaal bepaalde voor- en nadelen hebben.

### 2.2.1 Google

Een mogelijkheid om e-mails te sturen is om gebruik te maken van een al bestaande SMTP server. De meest gebruikte service is Google SMTP. Hiermee kan je tot 10.000 e-mails sturen per dag. Het is ook veilig om te gebruiken, en relatief makkelijk op te zetten. Deze service is gratis alleen niet open source.

### 2.2.2 Sendinblue

Een andere optie is Sendinblue. Hiermee kan je per dag 300 gratis e-mails sturen. Een andere optie kost 25 USD per maand om maximaal 20.000 e-mails per dag te kunnen sturen. Deze service is helaas niet open source.

### 2.2.3 Zelf hosten

Het nadeel van gebruikmaken van een externe server is dat je andere bedrijven moet vertrouwen om veilig met de data van je klanten/gebruikers om moet gaan, en bijna altijd niet open source is. Het is dus ook een optie om zelf een email server te hosten. Nadelen hiervan is dat je zelf de server moet onderhouden, wat soms veel tijd en geld kan kosten. Een veel gebruikte mailserver is Docker MailServer. Deze kan ook gebruik maken van de meest gebruikte mail protocollen en services.

### 2.2.4 Webhosting

Vaak is bij gekochte webhosting service naast het hosten van een webapplicatie, ook een mailserver inbegrepen. De opdrachtgever Ontdekstation 013 heeft een website, en een mogelijkheid is dat de opdrachtgever al een mailserver hosting oplossing heeft, waarvan gebruik gemaakt kan worden.

### 2.2.1 Conclusie e-mail

We hebben begrepen dat het Ontdekstation zo veel mogelijk waar het kan gebruik wil maken van open source opties, en denken dat of het zelf hosten van een mail server, of de eventueel inbegrepen mail server bij het webhosting de beste opties zijn.

## 2.3 Specificaties

### 2.3.1 Verificatie type

Voor het verifiëren van een email adres of telefoonnummer kan er gebruik worden gemaakt van een code die verzonden wordt naar het specifieke email adres of telefoonnummer die de gebruiker heeft ingevoerd. Het is ook mogelijk een URL-link te gebruiken in een email/sms om het desbetreffende te verifiëren. Dit is ook gebruiksvriendelijker aangezien er maar 1 knop in hoeft te worden gedrukt daarvoor, ten opzichte van het overtypen van een code. Maar dat is alleen handig als het gekregen verificatie bericht op hetzelfde apparaat wordt ontvangen als waarop de applicatie wordt gebruikt. Voor het verifiëren van een email adres is dit soms bij online-websites het geval, en voor telefoonnummers bij telefoon applicaties, aangezien het dan vanzelfsprekender is dat de gebruiker al gebruik maakt van het juiste apparaat.

### 2.3.2 Veiligheid

#### Cijfers of letters

Met het gebruik van een code kunnen er verschillende karakters gebruikt worden voor het verifiëren. Hierbij speelt de hoeveelheid achtereenvolgende en kwantiteit aan karakters en rol, met hoe meer, hoe veiliger. Gebruik makend van allebei cijfers en letters is er een grotere hoeveelheid, en dus veiliger, maar dit is minder gebruikersvriendelijk. De grote bedrijven Google en Microsoft sturen een 6-cijverige code voor het verifiëren van een telefoonnummer. Met zes cijfers zijn er 1.000.000 (10^6) hoeveelheid combinaties, wat meer dan genoeg is.

#### Vervaltijd

Bij het sturen van een code is het veilig om deze te laten verlopen na een bepaalde hoeveelheid tijd. Gebruikelijk is 24 uur om deze code geldig te laten blijven.

### 2.3.3 2-factor authentication

Na het verifiëren van de contact gegevens van de gebruiker is het een mogelijkheid om 2fa voor de gebruikers aan te bieden. Het is niet gebruiksvriendelijk om elke keer als de gebruiker inlogt een code te sturen, en dus zou dit beter op basis van locatie, bepaalde aantal tijd sinds de laatste login, of een andere factor kunnen worden bepaald.

Toepassing

## 3.1 Email toepassing

Voor deze toepassing willen we een demo applicatie maken in het Java springboot framework, waarbij de applicatie een e-mail stuurt naar e-mailadres die een code meegeeft.

In de eerste fase zullen wij bij het testen gebruik maken van Google ’s SMTP(Simple Mail Transfer Protocol) service. We maken hier tijdelijk gebruik van om op een simpele manier te testen hoe we mails kunnen maken en sturen met Java sprintboot. We verwachten dat we redelijk makkelijk een andere mail server kunnen gebruiken. De reden waarom we gebruik maken van Google mailserver is omdat we denken dat dit het makkelijkste gaat, en er de meeste informatie erover is op het internet in combinatie met het gebruikte framework.

### 3.1.1 Eerste email fase

Het is ons gelukt om op een simpele en effectieve manier mails te sturen door middel van Java en sprint boot. Dit doen we door middel van 3 bestanden.

#### Gebruikte bestanden

* Application.properties
* EmailSenderService.java
* JavaGmailSmtpApplication.java

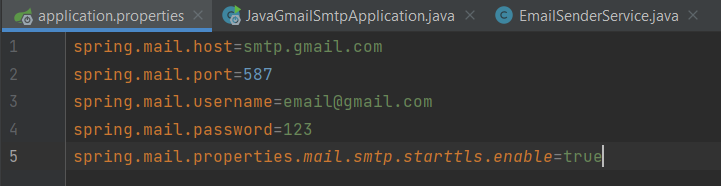
### 3.1.2 Uitleg bestanden

#### Application.properties

In dit bestand zetten we proporties die nodig zijn om de verbinding te maken met de SMTP server.

Zoals je hieronder kan zien, zetten we de volgende gegevens:

* Host | hiermee refereren we dus naar de smtp server van google.
* Port | Dit is de port die we moeten gebruiken om verbining te maken met de smtp server van google.
* Username | hier vullen we het email in waarvan we de mails gaan sturen aka de afzender.
* Password | Dit is het wachtwoord dat bij de email hoort. Als je dit niet heb kan je geen gebruik maken van de mail
* Starttls| Door dit op true te hebben zeggen we dat we gebruik willen maken van een tls (Transport Layer Security)



*Figuur 1*

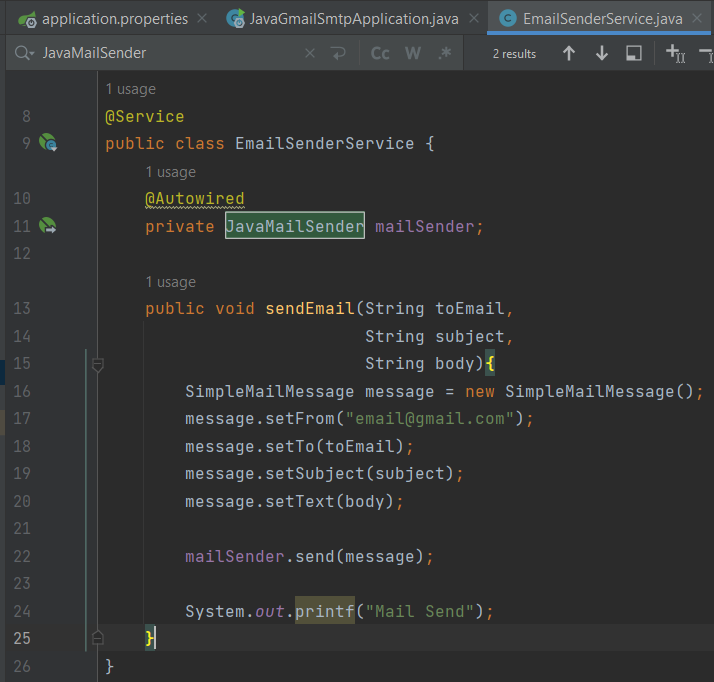
#### EmailSenderService

In dit bestand zetten we de functie voor het sturen van de data klaar.

We beginnen met JavaMailSender klaar te zetten door middel van @Autowired. Hiermee injecten we de geïmporteerde package van JavaMailSender op de private JavaMailSender.

Daarna zeggen we dat sendEmail vanuit [email@gmail.com](mailto:email@gmail.com) gestuurd word, een email adres nodig heeft om de mail naartoe te sturen, het onderwerp en als laatst de tekst die we in de mail willen.

Als laatst zeggen we dat de hele message door gestuurd word aan de ontvanger die we erboven in hebben genoteerd.

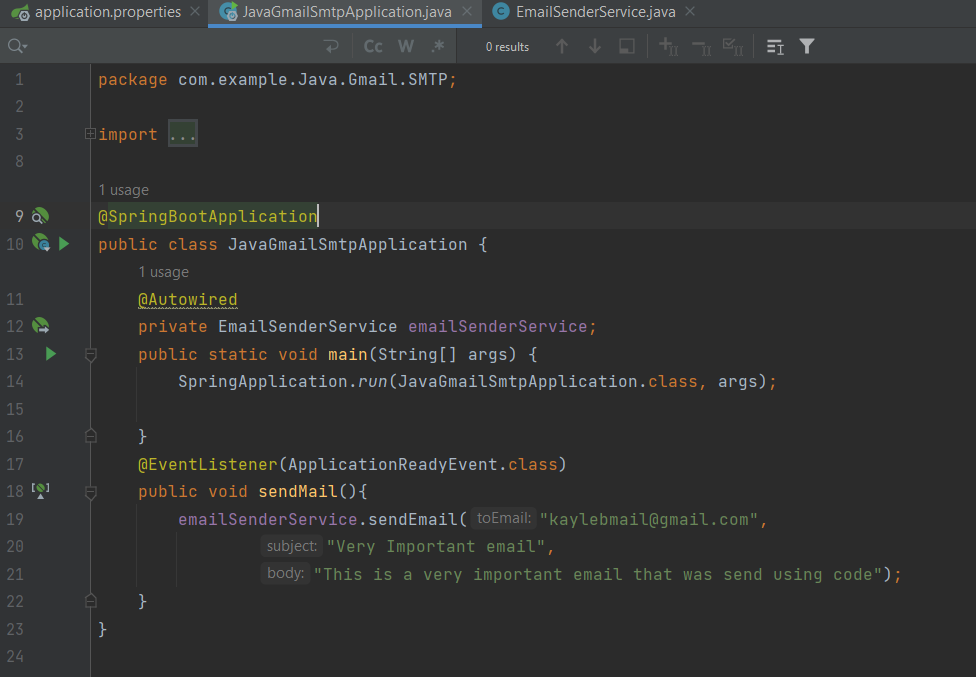


*Figuur 2*

#### JavaGmailSmtpApplication

Dit is eigenlijk ons hoofd bestand. Zoals je ziet op regel 11 halen we hier dus door middel van @Autowired het andere gemaakte bestand (EmailSenderService) hier op.

Als we dan op regel 17 kijken zien we dat wanneer de applicatie klaar is, de sendMail functie zal runnen. Hier vullen we dan ook de data die in de mail terug gevonden kan worden.



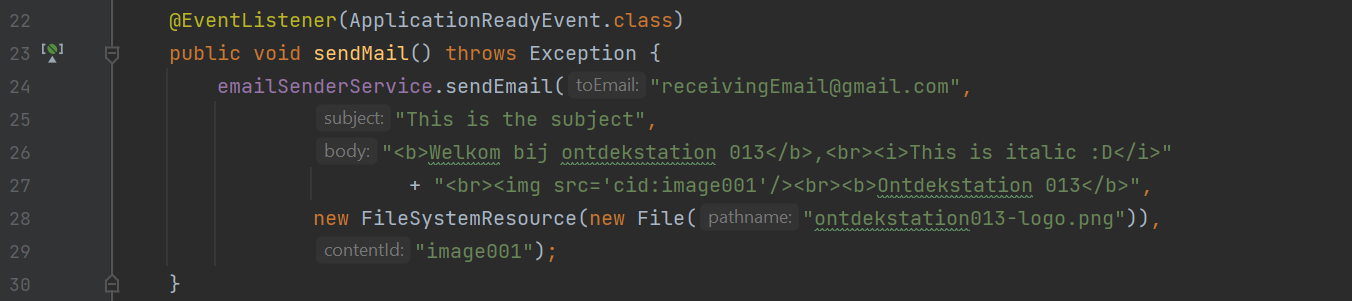
*Figuur 3*

### 3.1.3 MIME standaarden

Bij JavaMailSender kan er gebruik worden gemaakt van MIME standaarden. MIME (Multipurpose Internet Mail Extensions) is een internet standaard dat email uitbreidt, zodat er naast simpele ASCII, ook meerdere tekstcoderingen kunnen worden gebruikt, en er bijlagen kunnen worden toegevoegd.

#### Mime Messages

Bij het sturen van Mime Messages wordt het variabel message i.p.v. “SimpleMailMessage()” een object van het type “mailSender.createMimeMessage()” . Dat object wordt vervolgens gevuld door gebruik te maken van een “MimeMessageHelper”. Net zoals bij het SimpleMailMessage wordt er een “setText” functie gebruikt om data in de body van de email te plaatsen, alleen nu wordt er gebruik gemaakt van HTML, vandaar dat de tweede parameter in regel 27 op "true” staat. (*Figuur 5)* Zo kan je bijvoorbeeld tekst bold of italic maken, en bijlagen toevoegen aan een email. In dit voorbeeld is er een inline image toegevoegd aan de tekst, door in de HTML string een <img> tag toe te voegen met als de source een "contentId”, in dit geval “image001”. Vervolgens met de “helper.addInline” functie wordt de image inline geplaats in de email tekst.

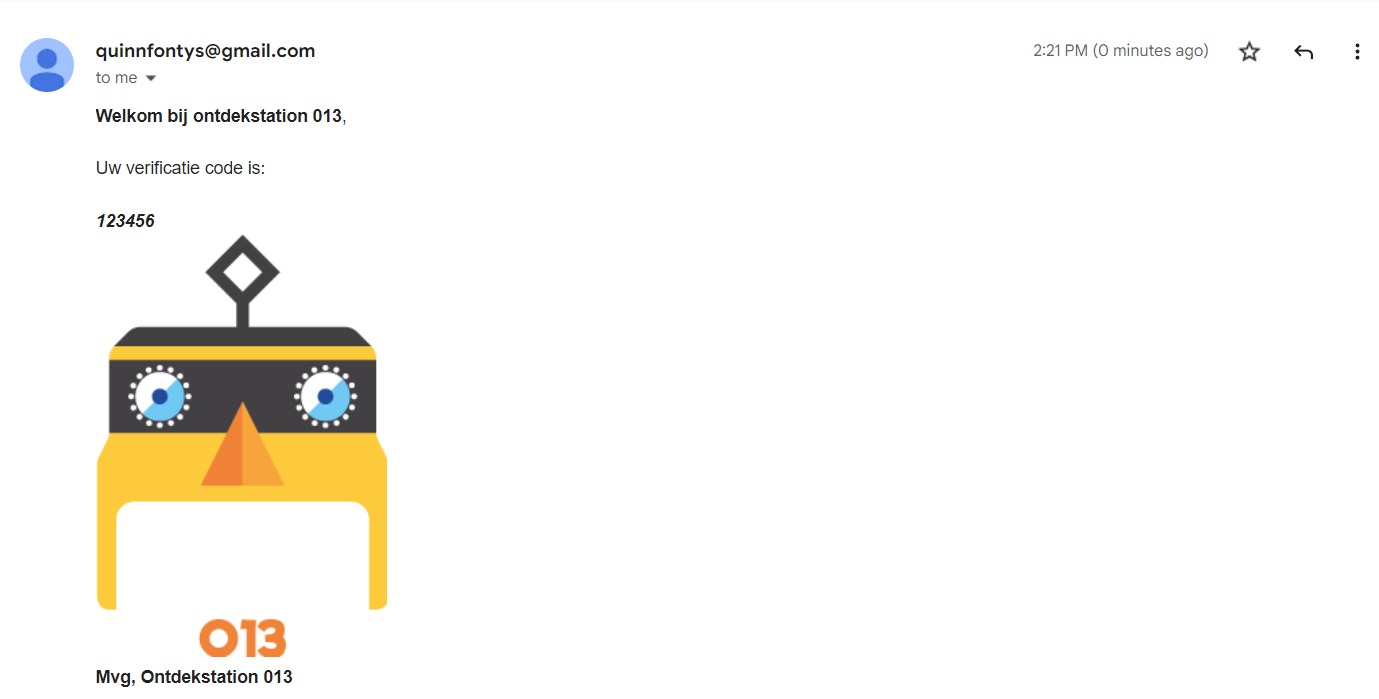
*Figuur 4 (Functie in EmailSenderService bestand)*

*Figuur 5 (Functie in JavaGmailSMTPApplication bestand)*

Iets om op te merken is dat allebei de functies “SendMail()” en “EmailSenderService()” geen unhandled exceptions kunnen hebben vanwege de MimeMessage en MimeMessageHelper classes, en daarom een exception handler toegevoegd moeten hebben.

### 3.1.4 Samenvatting

Nu zien we dat we eigenlijk op een hele korten en duidelijke manier een email kunnen versturen met behulp van Java. Natuurlijk zouden we de emails, onderwerpen en body's makkelijk aan kunnen passen waar dit nodig is zodat het niet hard coded is. Gebruik maken van MIME berichten voor een beter uitziende email is qua gebruikersvriendelijkheid veel beter.

*Figuur 6*

Dit demo project is compatibel om in het bestaande project verwerkt te worden.

## 3.2 Verdere uitwerking

Voor de gestuurde E-mail of SMS zou een willekeurig code moeten worden gegenereerd, en tijdelijk opgeslagen worden in de database. Vervolgens als de klant de code eenmaal heeft ingetypt in de webapplicatie wordt het vergelijkt of het overeen komt met de desbetreffende gebruiker op dat bepaald tijdstip. Ten slotte is het telefoonnummer of email adres geverifieerd, of juist niet.

# Conclusie

Wij zijn van mening dat alhoewel het gebruiksvriendelijker is om de klant de optie te bieden tussen contact krijgen via SMS of E-mail, het toch niet waard is om de SMS optie aan te bieden, aangezien de verwante kosten. We hebben vernomen dat het Ontdekstation zo veel mogelijk waar het kan gebruik wil maken van open source opties, en denken dat of het zelf hosten van een email server, of de inbegrepen mailserver bij het webhosting. Zoals in de toepassing laten zien en beschreven is het mogelijk om een email te sturen vanuit de applicatie.

# Sources

2.1.1

<https://www.twilio.com/blog/sms-spring-boot-app>

<https://www.baeldung.com/java-sms-twilio>

<https://www.pixeltrice.com/send-an-sms-or-message-from-spring-boot-application-to-mobile-phone/><https://www.tutorialspoint.com/spring_boot/spring_boot_twilio.htm>

2.1.2

<https://www.plivo.com/blog/receive-respond-sms-java-spring-plivo/>

2.1.3

<https://aws.amazon.com/sns/?whats-new-cards.sort-by=item.additionalFields.postDateTime&whats-new-cards.sort-order=desc>

2.2.1

<https://support.google.com/a/answer/176600?hl=en>

<https://www.hostinger.com/tutorials/how-to-use-free-google-smtp-server>

2.2.2

[https://www.sendinblue.com](https://www.sendinblue.com/)

2.2.4

<https://geekflare.com/self-hosted-email-server/>

3.1

<https://www.javacodemonk.com/spring-boot-send-email-with-gmail-smtp-5caea8f3>

<https://www.ibm.com/docs/en/b2badv-communication/1.0.0?topic=concepts-mime-multipart-message-overview>

4.3.2

<https://www.baeldung.com/spring-autowire>