

SIT313

Dev@Deakin

Welcome Email



**DEAKIN**  
UNIVERSITY



## SIT313 - Full-Stack Development: Secure Frontend Applications

### DevLink Marketplace Welcome Email

#### Overview

This task intends to provide you with experience in [Node.js](#), [Express](#) and [Web APIs](#). You are given the requirements of sending a welcome email to new subscribers or users who join the DEV@Deakin platform.

You will find "Demo Videos" of Week 2 Practical on the unit site to be particularly useful as a reference for this task. Please also keep an eye on your email and any announcements that may be made on Cloud Deakin.

#### Task Requirements

The DEV@Deakin application needs to send various types of transactional emails including welcome emails, password reset links and in-app notifications such as new messages. In this task, you need to develop a component to send a quick welcome message to new subscribers after submitting their emails as shown below.

A horizontal form element. On the left is a grey button with the text "SIGN UP FOR OUR DAILY INSIDER" in white. To its right is a white input field with a thin border containing the placeholder text "Enter your email". To the right of the input field is a grey button with the text "Subscribe" in white.

The welcome message will automatically go out to a new subscriber who joins the platform. You need to use Email APIs like [SendGrid](#) to send emails asynchronously from the DEV@Deakin application. Please check [SendGrid welcome email](#) for welcome emails and [here](#) is a good tutorial on how to set up. If you have any difficulties in creating a SendGrid account and facing issues with the verification process, you can use MailGun API or any other email service APIs for this task. [Here](#) is a step-by-step tutorial to guide you through the MailGun account setup process.

#### Submission Instructions

To submit this task on OnTrack, please follow the guidelines below:

1. Ensure that all program files related to this task are stored in a Git repository named "Task 2.1P".
2. Within the repository, create a subdirectory called "public" to house your CSS files and images. Place the CSS files under the "public/css" directory and the images under the "public/images" directory.
3. Remember not to include the "node\_modules" directory when pushing files to the GitHub repository. Create a .gitignore file in the root directory of your repository. Inside the .gitignore file, include the following line:



```
node_modules/
```

This will exclude the "node\_modules" directory from being tracked by Git.

4. Submit the GitHub repository link and all required files through the task submission page on OnTrack. Consolidate all links into a single PDF file before submitting.
5. It is essential to grant access to the GitHub repository so that the assessors can review your work.
6. Additionally, include a demo video of your webpage and a welcome email. Both the video and GitHub repository should not be public. If you choose to upload the video to platforms like YouTube or Deakin Air, ensure that it is set to private and only accessible to your teaching team.
7. Finally, submit only the "server.js" file to OnTrack.