

Project	Project Manager	Date
Scope document for Cron-email Utility	Ganesh Salunkhe	02/09/2024

Introduction
The Cron-Email Utility is a web-based application that allows users to schedule and manage emails using a user-friendly interface similar to Gmail. The utility integrates with nodemailer and npm-cron packages to facilitate seamless email scheduling and delivery.

Scope Description
In Scope
<ul style="list-style-type: none"> <li>User authentication: Sign-up, login, and logout functionalities.</li> <li>Email scheduling: Interface to schedule emails with options to select date, time, recipients, and email content.</li> <li>Email management: Ability to edit and delete scheduled emails.</li> <li>Integration: Use Nodemailer for sending emails and npm-cron for scheduling tasks.</li> <li>Security: All sensitive user data, including email credentials, must be encrypted.</li> <li>Usability: The interface should be user-friendly and accessible, similar to Gmail's design.</li> </ul>
Out of Scope
<ul style="list-style-type: none"> <li>Advanced email analytics or reporting.</li> <li>Mobile application development.</li> </ul>

Business Objectives
<ul style="list-style-type: none"> <li>Provide a dashboard for managing sender email information.</li> <li>Allow users to schedule emails to be sent at specified times.</li> <li>Enable users to edit or delete scheduled emails.</li> <li>Ensure the user interface is intuitive and similar to Gmail's layout.</li> </ul>

Dependencies
<ul style="list-style-type: none"> <li>Dependency on Nodemailer for email functionality.</li> <li>Dependency on npm-cron for scheduling tasks.</li> <li>Dependency on a database service (e.g., MySQL or MongoDB) for storing user and email data.</li> <li>Dependency on a json-web-token for user authentication.</li> <li>Dependency on a bcrypt.js for password hashing.</li> </ul>

Assumptions
<ul style="list-style-type: none"><li>• Users will have internet access to use the web application.</li><li>• The email servers used by the application will have sufficient capacity and permissions to handle the volume of emails.</li><li>• The email addresses provided by users are valid and can be used for sending emails.</li></ul>



Deliverables
<ul style="list-style-type: none"><li>• A functional web application with user authentication.</li><li>• A dashboard for managing scheduled emails.</li></ul>



Acceptance Criteria
<ul style="list-style-type: none"><li>• The application passes usability tests and meets performance benchmarks.</li><li>• All user stories and requirements are implemented and tested.</li></ul>