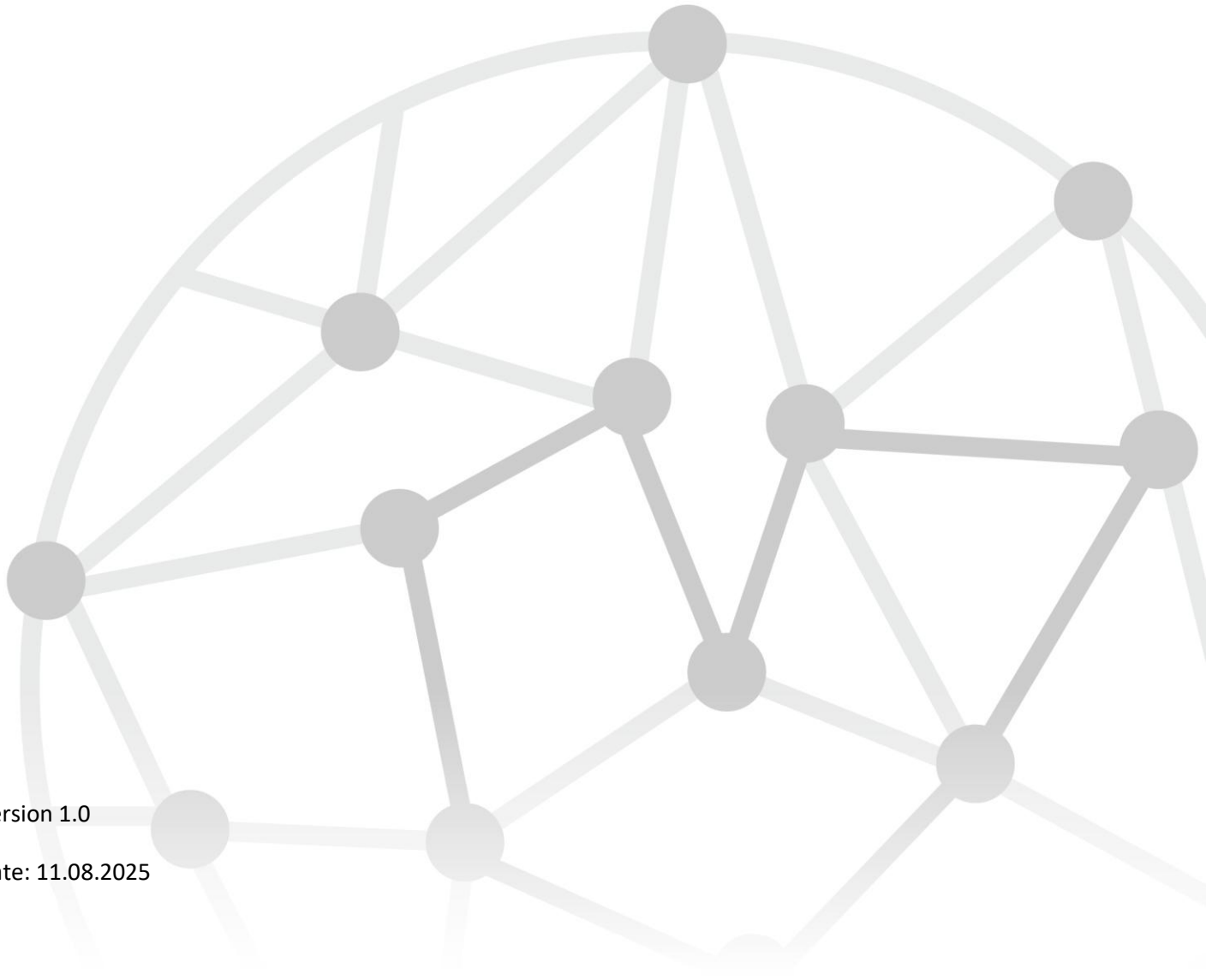
VESTEL

FROM IDEA TO DONE.

Entry Full Stack Developer – Technical Assessment Tasks



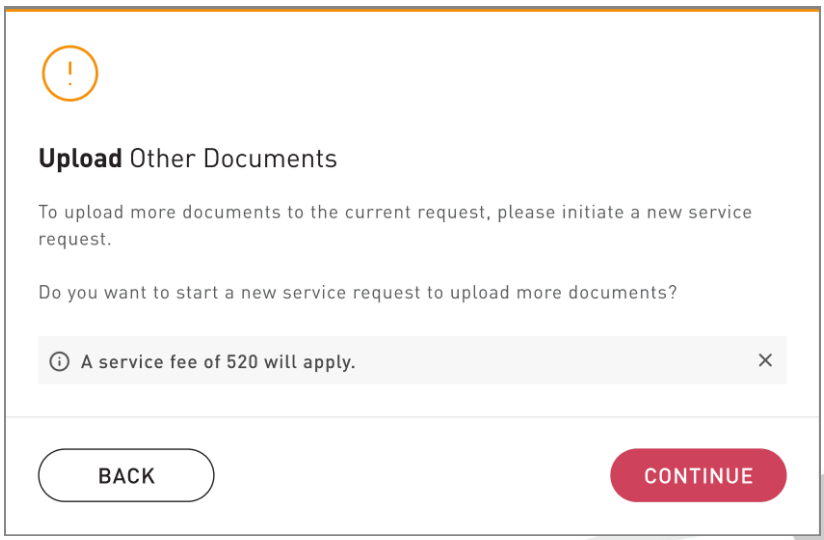
Version 1.0

Date: 11.08.2025

Task 1 – Frontend Component Implementation

Objective: Build a UI component based on the attached image, using HTML, CSS, and JavaScript.

Requirements: The component's layout and style must match the design below as closely as possible.



Use the following base CSS for the main container:

display: flex;

padding: var(--Spacing-3xl, 24px);

flex-direction: column;

align-items: flex-start;

gap: var(--Spacing-3xl, 24px);

align-self: stretch;

JavaScript should be used only if needed for component behavior (e.g., interactivity).

Submission: Provide the HTML, CSS, and JS files (or a single HTML file with embedded CSS/JS).

The component should be viewable in a browser without additional build steps.

Note: The functionality of the component is not important for this task — focus only on matching the visual appearance of the attached design.

Task 2 – .NET Core API

Objective: Create a small .NET Core Web API endpoint that accepts a text string and returns the reversed text.

Requirements:

- **Framework:** .NET Core 6 or later
- **Endpoint:** POST /api/reverse

Input:

JSON in the format:

{ "text": "hello world" }

Output:

JSON in the format:

{ "reversedText": "dlrow olleh" }

Validation:

If the text property is missing, empty, or null → return HTTP 400 with a message:

{ "error": "Text cannot be empty." }

Bonus (optional):

Add a unit test that checks the reversal logic.

Submission: Provide the complete .NET Core project in a .zip file or share a link to a public GitHub repository.