**Infrastructure Overview**

A computer screen with a black screen and green squares

Description automatically generated

**Application Layer (UI, API)**

* Handles user interactions and API requests.
* Built with .NET, serving as the gateway for client requests.
* Communicates with the Core layer and sends tasks to RabbitMQ for asynchronous processing.

**Core Layer (Domain, Services, Repositories)**

* Shared logic layer for the Application layer and Worker nodes.
* Manages domain models, business services, and data operations.
* Ensures consistent and reusable business logic.

**Worker Nodes**

* Background processing units using .NET background tasks.
* Consume and process tasks from RabbitMQ.
* Auto scaled by Kubernetes based on workload demands.

**Storage**

* Shared data storage accessible by Application and Worker nodes.
* Ensures seamless data consistency and retrieval.

**Use Case: Email Campaign Processing**

1. A user triggers an email campaign via MailCrafter UI.
2. The Application layer validates the request and queues the task in RabbitMQ.
3. Worker nodes consume tasks, retrieve templates, personalize content, and send emails.
4. Both Application and Worker nodes interact with storage for data retrieval and updates.

**Serverless Implementation**

A screenshot of a computer

Description automatically generated