



## Technical description

### Introduction

**Project name:** Oncological Reports Capturing Application (ORCA)  
**Purpose:** Capturing, management and evaluation of tumour reports  
**Authors:** Manuel Jordan  
**Creation date:** 12.08.2024

### Technology stack

**Frontend:** Blazor Server  
**Backend:** .NET 8, ASP.NET Core  
**Database:** PostgreSQL  
**Other Technologies:** Entity Framework, SignalR

### Third-Party-Libraries

**Tesseract OCR Engine:** <https://github.com/charlesw/tesseract>  
**Blazored:** <https://github.com/Blazored>  
**Syncfusion Library:** <https://www.syncfusion.com>  
**Mudblazor:** <https://mudblazor.com>

### System requirements

**Hosting:** Docker / Kubernetes / Windows-Server  
**Resources:** 2 CPU, > 4GB RAM  
**Document storage:** Fileshare oder S3

### App components

**ORCA-App:** Frontend, File-Importer, Mail-Importer  
**ZAS-Service:** ZAS query for AHV / OASI number determination  
**OCR-Service:** text recognition  
**Nicerstat-Service:** Tumor synchronization  
**API-Service:** REST API for uploading documents  
**RabbitMQ:** Messaging-Broker

### Other features

**Authentication and Authorization:** local user accounts  
**App configuration:** config.json file  
**Logging:** separate, rotating log files via Serilog