

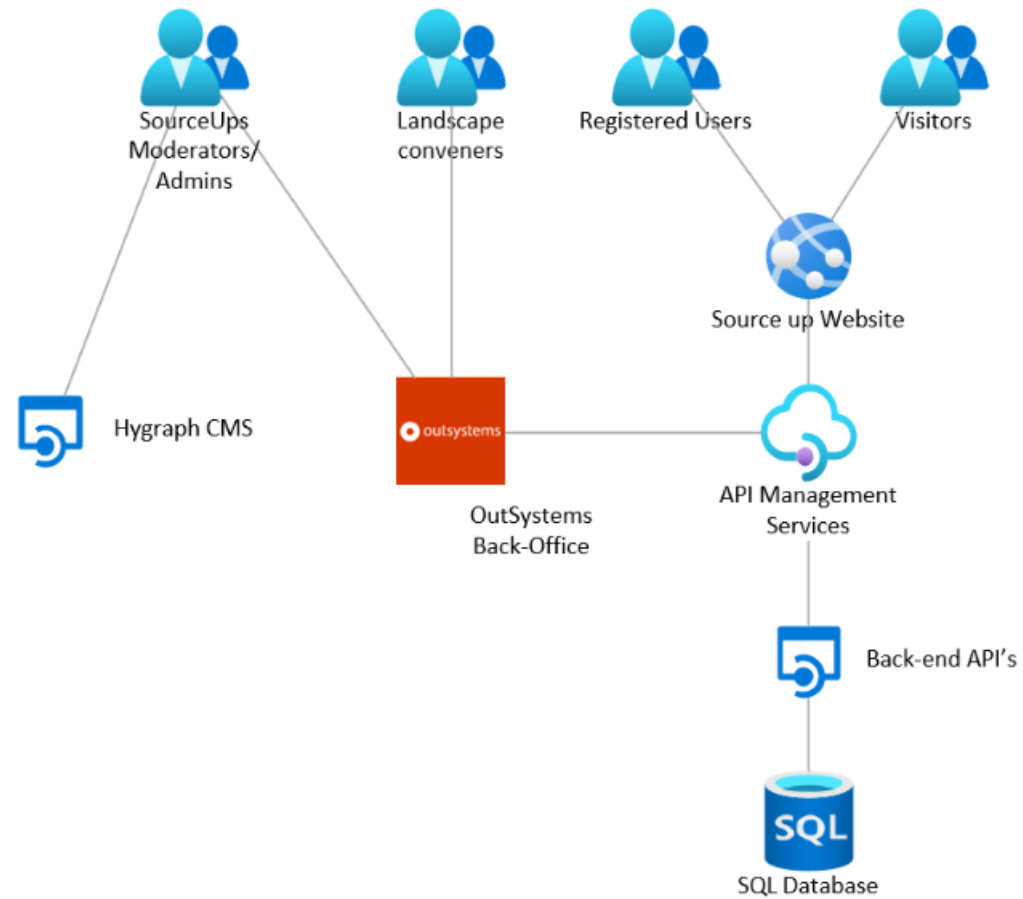
Annex 3 – Tech Stack

1. SourceUp Platform technology stack overview¹

Category	Technology	Version (where applicable)	Relationships/Responsibilities
Hosting and infrastructure	Terraform	0.14	Infrastructure as code, allows or consistent infrastructure provisioning and configuration.
Hosting and infrastructure	Microsoft Azure App Services	N/A	Hosting API's and Front-end
Hosting and infrastructure	Microsoft Azure SQL	N/A	Hosting Databases
Hosting and infrastructure	Microsoft Azure Application Gateway w/ WAF	N/A	Firewall/load balancing
Hosting and infrastructure	Microsoft Azure Storage	N/A	Generic storage for monitoring data
Hosting and infrastructure	Microsoft Azure CDN	N/A	Content delivery for improved performance
Hosting and infrastructure	Microsoft Azure AD B2C	N/A	Authentication platform
Back-end microservices	.NET Core	3.1	Framework used to build the back-end API
Back-end microservices	Open API Specifications	3	Used to generate API documentation and import into Azure API management
Back-end microservices	Nunit	3	Unit testing framework
Front-end application	TypeScript	5.2	Programming language used in the front-end
Front-end application	ReactJS	18	Part of front-end framework
Front-end application	NextJS	13.5	Part of front-end framework
Front-end application	Redux	4.2	Used to manage state in front-end
Front-end application	SASS	1.67	Front-end styling framework
Front-end application	Jest	29.7	Front-end testing framework
Front-end application	Axios	1.5	Used to execute HTTP Requests from the frontend
Front-end application	Apollo	3.8	Part of front-end framework, state management.
Content Management System (front end)	Hygraph	N/A	Content management system. Used by the front-end to fetch and display content
Content Management System (backoffice)	Outsystems	11.54	Used to manage content of business entities, like Compacts.
Content Management System (front end + backend)	Azure AI Translator	N/A	Used to translate content.

¹ The versions in the document reflect the current versions that are being used in the platform. However, some components need to be upgraded.

2. High-level architecture diagram of the main tech stack components and their relationship.



3. **Tech components environments** (development, test, acceptance and production)

The components listed below have a development, acceptance and production environment:

- SQL Databases
- Back-end API's
- Front-end application (source up website)
- OutSystems

Hygraph CMS only has two environments; development and production. The Hygraph development environment is being used by the development and acceptance environment of the front-end application. A third environment is only available by upgrading to the next pricing tier, which is a significant increase in costs. This is the reason why we are using just two environments.

Programming languages used in the back-end are Back-end C# .NET, SQL and if the front-end are Typescript. In Outsystems no code is being written.

4. **Hosting of Tech components**

Most components are being hosted in Microsoft Azure cloud services. Outsystems and Hygraph are being hosted in their own respective cloud providers. The responsibilities of maintaining the hardware and rolling out (security) updates for all hardware and software necessary for hosting, is outsourced to the respective cloud providers. In the case of Microsoft Azure, we are responsible for provisioning the needed services within azure and configuring them accordingly to run the software we deploy to them.