Emgage Web Services Overview

Emgage Web Services provide a powerful development environment that allows individuals to drag and drop app components, customize them, connect them, and create unique business software solutions. Whether you are an IT professional, an entrepreneur, or even a non-tech-savvy person, you will have the means to create your software products effortlessly. Our low-code no-code approach allows developers to build, integrate, and deploy applications without having to write code line by line.

Emgage offers a user-friendly interface (UI), through which you can link multiple SaaS applications and automate processes – all within the same platform. Instead of worrying about the tools to create and maintain their software, our users will now be able to concentrate on their core business needs in full. Go above and beyond with Emgage Web services, where we help build your apps, test them, host them, and even sell them in our marketplace.

Main Adavantages of Emgage Web Services

1. Help IT departments, developers, and entrepreneurs to build B2B & B2C software solutions
2. Users do not need high technical skills to create and maintain software
3. Minimize security vulnerabilities such as data loss/breach, unauthorized access, and malware
4. Solves SaaS integration issues, bringing an overall enhancement in the business applications
5. The platform's ease of use reduces current SaaS over-budgeting issues
6. High capacity and flexibility compared to other low-code no-code solutions

> Why is this Important?
>
> > The typical business with 200 to 500 people (medium size) employs approximately 123 SaaS applications, according to the New York-based startup Blissfully. Around 40 applications are used by companies with 50 or fewer employees, compared to nearly 200 apps integrated by those with over 1000 employees. As the world of technology advances, more integrations between applications will be needed, which can translate into budget, management, and software compatibility issues.

Three ways of using our services:

Emgage Service	**Benefits**
App Builder	Drag & Drop UI to build business applications
Hyperconsole ™	Write code (Javascript or Curl) to personalize and manage your apps

Emgage Service	Benefits
Marketplace	Find apps that others have built and use them, or publish your own apps and generate revenue

Current enterprise use cases of Emgage Web Services

- Vendor Management
- Cybersecurity Compliance
- COVID Symptom Cert
- Facility Management System
- Georgia Housing Provider
- Shrimp Farming IoT App

Example

Creating a Vendor Management Application

- This is an overview of the Emgage Applications that can be used to fulfill today's business software requirements

Functionality	Emgage Application
Database to store the vendor's information	Data & Model
Provide the users with a login	Login & Authentication
Use documents to upload applications	Documents
Message and notification system	Messages
Online signature access capability	eSignatures
Approval workflow for new vendors	Workflow

> **Important:** Every single fuctionality of a standard business App, can be added with the Emgage Web Services environment

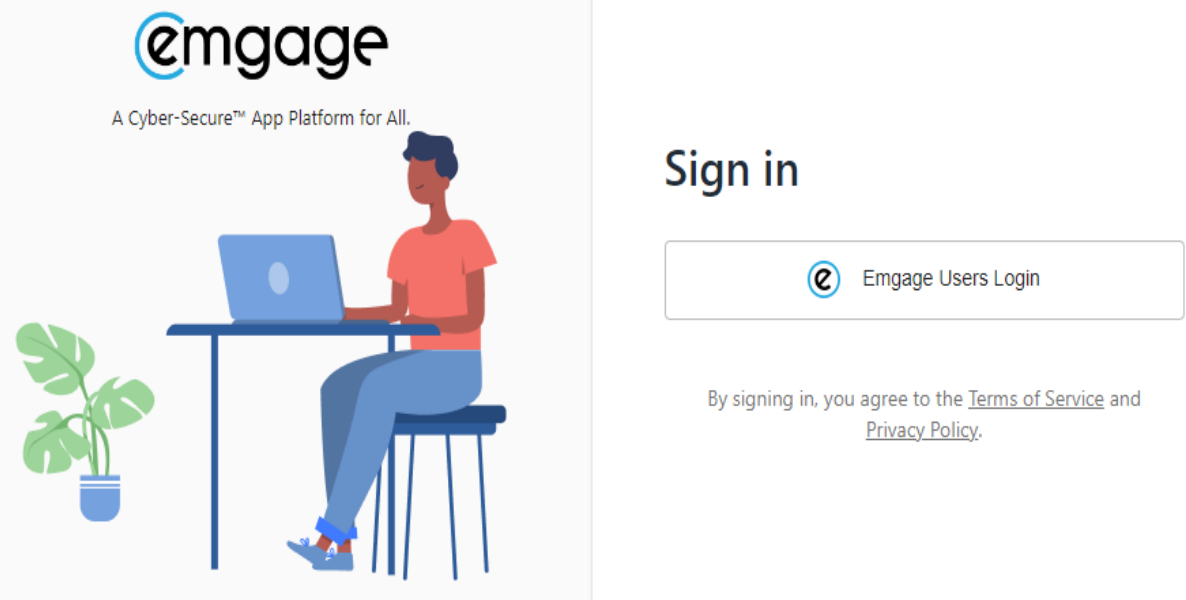
Step by Step: Create a Shell App

Emgage Web Services simplify the way you can develop business applications. This document describes how to create a shell app in Engage.

Before you begin

1. Go to the Emgage Sing In Page
    - An **Emgage Users Login** will be provided by default

> **Important:** You will be able to set up your own single sign-on with a personalized login

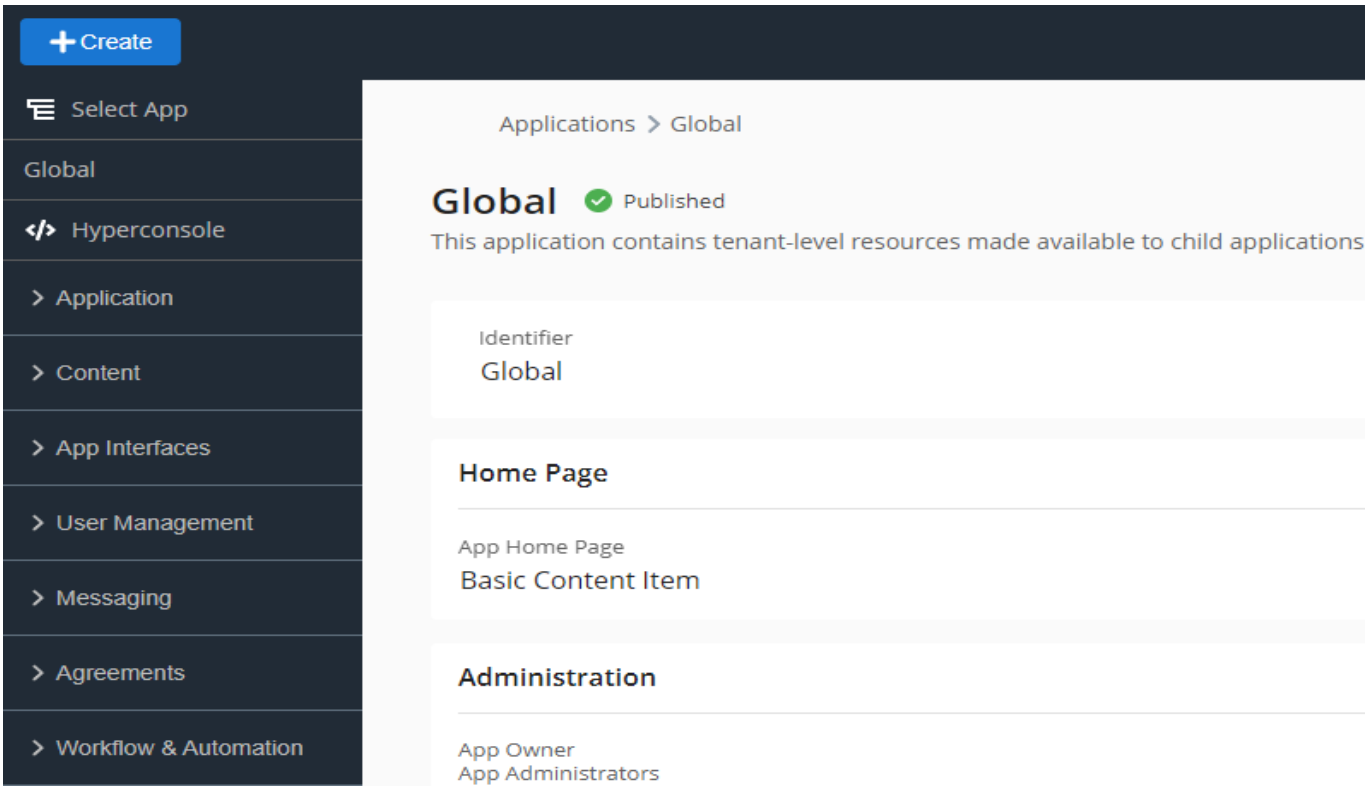


- After logging in, the *App Builder* will show up on the main page of the website.

Creating the Shell App

2. Click **Select App** on the top left corner of the menu to access the applications

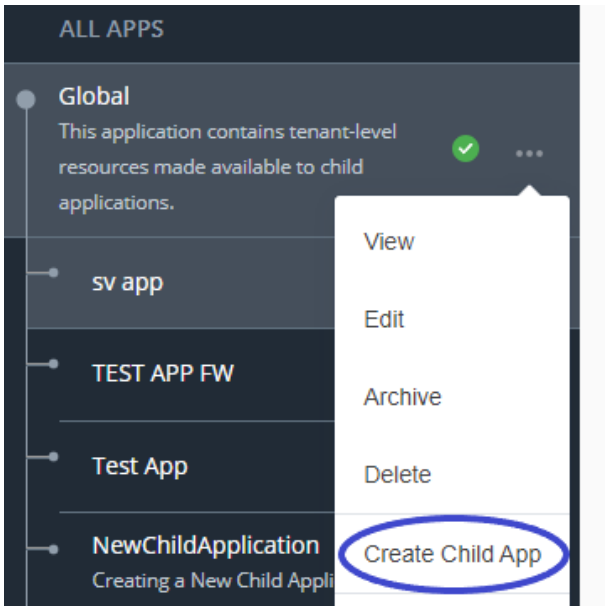
> **Important:** This is a test development environment



- The left side panel will display the Hyperconsole, as well as the embedded functionalities and services within the App Builder

---

3. Select the **Global** App on the top left corner
    - Click on the horizontal ellipsis (three horizontal dots)
    - Click on **Create Child App**

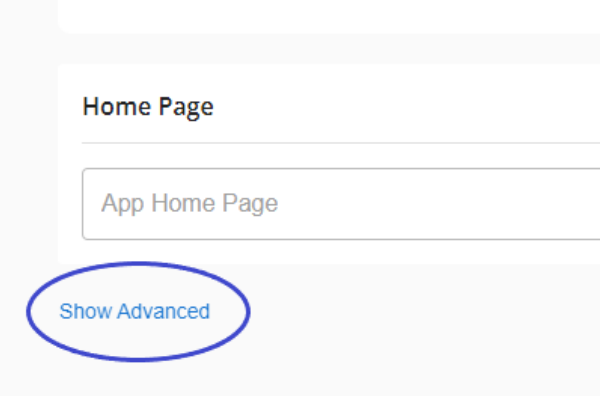


- Fill out the format **Create New Application** with the App description
    - An *identifier* will be generated automatically to access the App
    - Provide a detailed description for your App
    - Pick any **Home Page** option to set as default

> **Tip:** By providing accurate information, you can help keep your App secure and make our services more useful

---

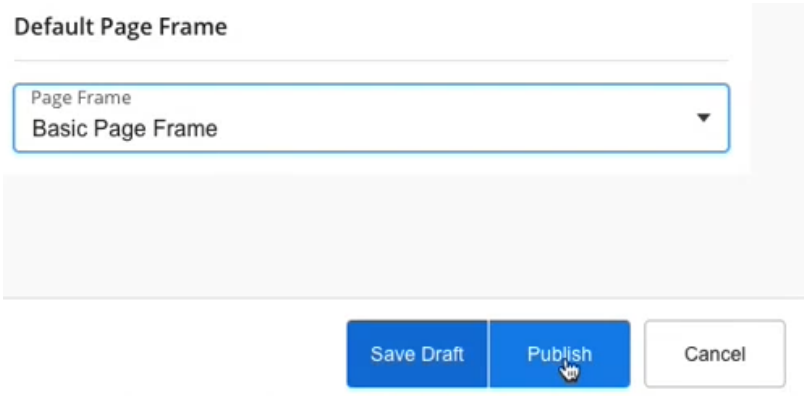
4. Click on **Show Advanced** settings
    - Fill out the blank spaces with your App information
    - Select the preferred options available for your App settings
    - The **Page Frame** section at the bottom will be set by default



> **Important:** Go to the **File Policy** and **File Store** sections to learn more

---

5. Click on **Publish** to make your App available for everybody
    - An *empty* App will be created

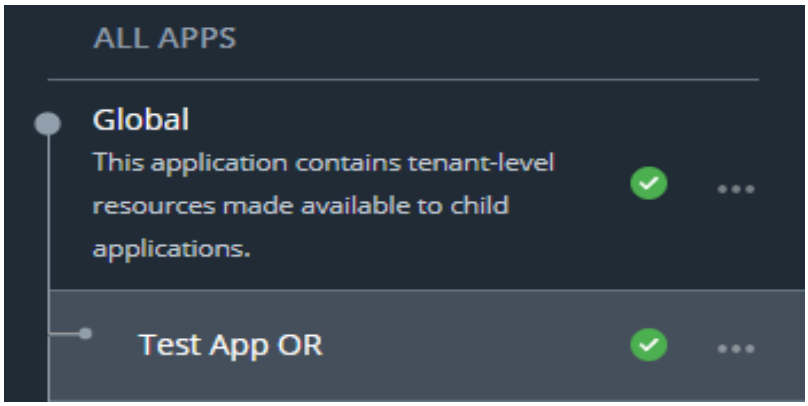


> **Important:** Go to the **File Policy** and **File Store** sections to learn more

What's Next

The new **Shell App** is the Child of the **Global** App

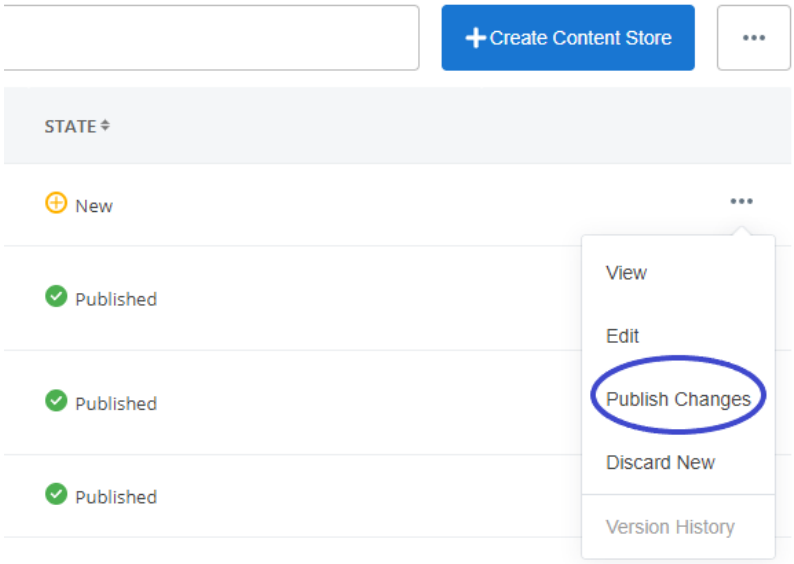
- All the services available for **Global** will be inherited by the new App, including users, groups, workflows, dashboards, permissions, pages & forms, and identity management, among others.



Add a new functionality inside our existing App

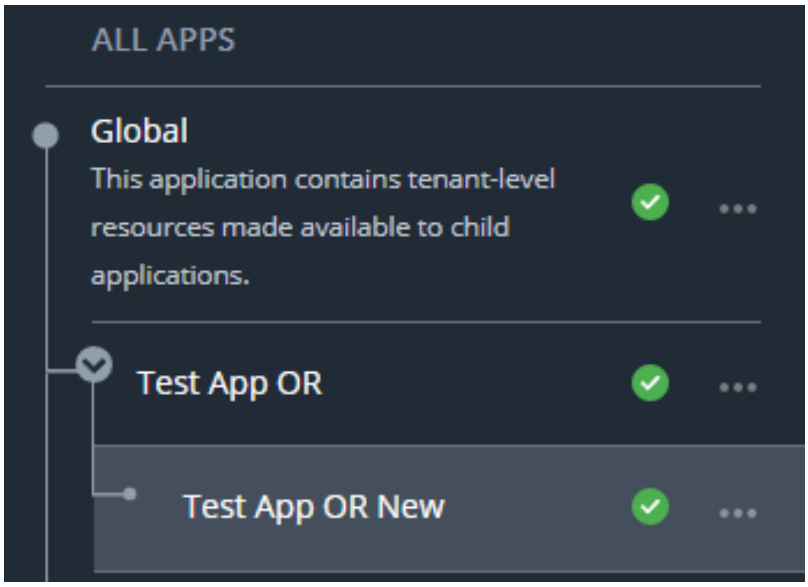
1. Select the **Shell App** you just created

- Go to the left side panel
- Click on the new App (*or functionality*) you want to add to your existing App
- Fill out the blank spaces with the information required
- Close the new App you just created
- Click on the horizontal ellipsis (three horizontal dots) & **Publish Changes**



---

2. Select the **Shell App** you created at the beginning
    - Click on the horizontal ellipsis (three horizontal dots)
    - Click on **Create Child App**



There are hierarchy levels to these applications, and they can inherit capabilities from each other

> **Important:** You can create as many **Child Apps** as you want, by following steps 3 to 5. Every new app will keep inheriting services and functionalities from the **Parent App**. This feature adds value to the software integration.

Term Glossary

App Builder

An online tool that helps create, customize, and optimize software solutions for individuals, companies, and governments. It uses graphical tools with embedded functionalities to reduce traditional code-writing requirements.

Child App

An application created on the grounds of a previous (*parent*) app, which inherits all the capabilities of the first one. It serves to organize and manage the internal processes of software development.

Hyperconsole

An alternative service platform within the Emgage environment, that allows developers to modify their applications through basic and simple code writing.

Low-code No-code

Unconventional application development that does not require the use of highly skilled code-writing practitioners. It is driven by automatic code generation, that leverages intuitive visual programming.

Parent App

The first application that is created within the emgage environment, from which new multiple apps can be developed. It contains the main services and functionalities that could be inherited to further software (*child*) applications.

Test Environment

An online space that allows users to run and validate experimental software cases within the Emgage platform.