Pega Express™ is a light, design-focused methodology that uses Pega’s low code experience, best practices, and scrum to deliver meaningful client business outcomes quickly. Pega uses design-thinking techniques to break customer journeys into smaller, more manageable pieces called Microjourneys™.

**Low Code Mean**

the low-code development tools create the code for you. User interface capabilities such as drag-and-drop, process flows, and visual tools allow anyone, regardless of technical ability, to create transformational software. This approach enables increased productivity, as everyday app development tasks are streamlined, lessening IT involvement. Low-code tools make application development simpler.

**Pega Express Methodology**

Pega Platform™ applications drive and facilitate customer interactions.

The Pega Express™ methodology is an agile approach that uses design thinking practices to capture the customer journey and quickly deliver a Minimum Loveable Product (MLP) release.

The Pega Express methodology breaks the customer journey into smaller pieces called **Microjourneys**™, which drive the organization to achieve a specific goal for the customer.

MLP release catagories into four

1. Microjourneys
2. Personas & Channels
3. Data & interfaces

**Microjourneys**

**Small part of customer, specific goal for customer**

For example, a customer wants to change their address. This scenario is a Microjourney that starts with the customer's request and results in an outcome, where their address is changed in the company's records.

**Personas & Channels**

Personas -> who interacts with application

Channels -> individuals interacts with application

for example, a persona could be a customer or a company employee, and a channel could be a web portal or a chatbot. An application can have multiple personas and multiple channels.

**Data & interfaces**

**Data** is the information that the Microjourney interacts with to accomplish the customer's goal, and the **interface** defines where the data comes from or where it is persisted. An application can interact with multiple types of data,and data can employ multiple interfaces.

For example , customer address is data and system which information resides is interface

Persona 🡪 customer

Channel 🡪 customer web portal

Data 🡪 Mailing address

Interface 🡪 Postal service

**App Studio**

Pega platform studios

Workspace is an environment to provide specific tools and featured.

App Studio

Dev Studio

Predicition Studio

Admin Studio

App Studio -> build an app without touching line of code

Dev Studio -> developers to go deeper into pega platform

Prediction Studio -> Built-in decisioning & AI powerup app build by pega

Admin Studio -> Manage security, devops, users & cloud performance of app

**Applications Layers**

Pega Platform applications are made up of instructions, called rules,

**Application Layers** section, which is a visual representation of the rules that make up the application. Layers define where rules reside and where rules are maintained.

**Role-based workspaces**

In Pega Platform, you can improve productivity by using role-based workspaces. Users see what they need when they need it.

Users can have access to multiple workspaces. When users log in, the system opens the default workspace. Users can access multiple workspaces and have the ability to transition between different studios.

**Dev Studio**

In App Studio, domain experts can access core application development features (case design, data management, and user experience) and apply their knowledge to improve development outcomes. Dev Studio provides features for configuring security permissions and access control, managing rules to promote reuse, and addressing the performance limitations of an application

### Naming differences between studios

App Studio Dev Studio

Field Property

Goal and deadline Service Level Agreement (SLA)

Process Flow

App Studio 🡪 citizen developers, front end designer, business analyst

Dev Studio 🡪 Database administrator, full stack developer, security admin

Case Life Cycle

Submit order 🡪 process order 🡪 deliver order

Case types and cases

Case type is an abstract model of business transaction.

Case is a specific transaction instance

## Service-level agreements

## A ****service-level agreement (SLA)**** establishes a work completion deadline.

## Urgency

Urgency is a numeric value that brings visibility to unresolved work in your application. An assignment has a default urgency of 10.

### **Assignment priority**

In Pega applications, assignment urgency indicates assignment priority.

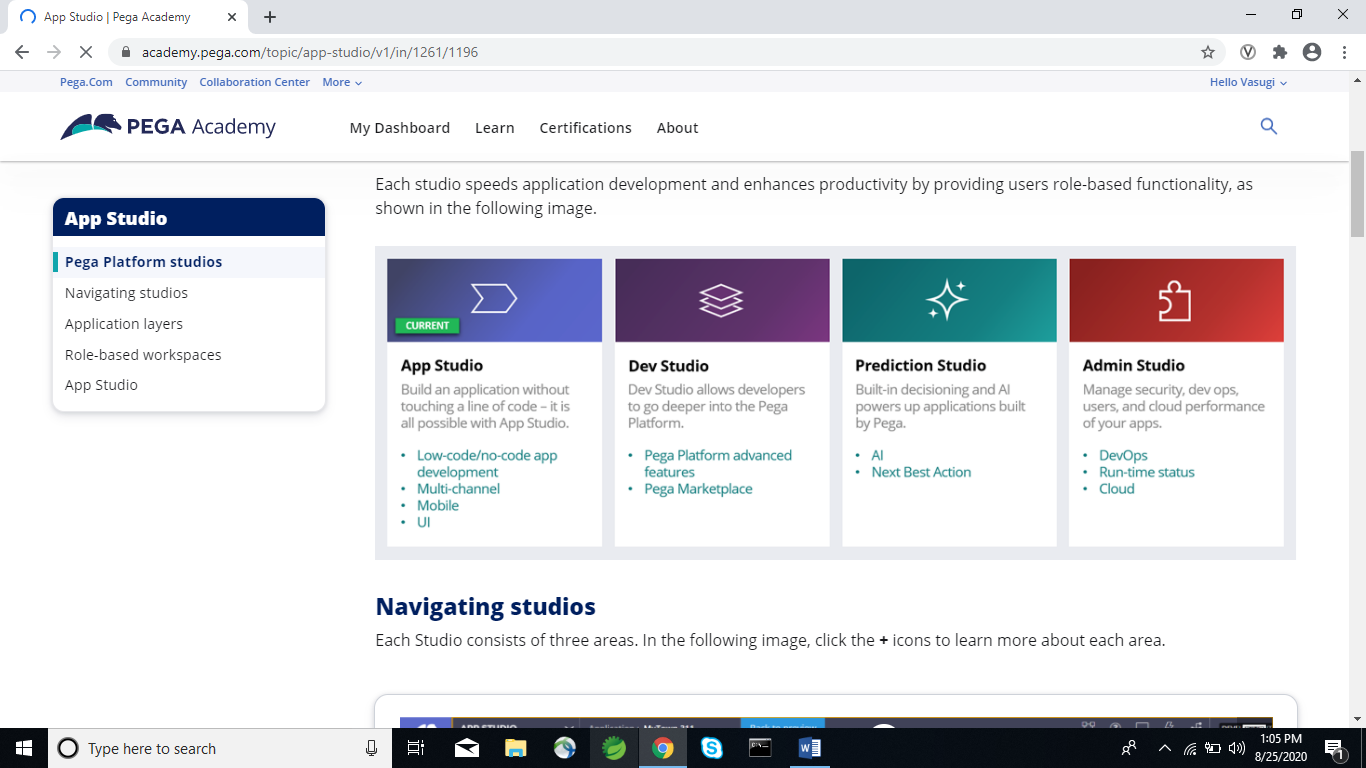
**Case Approvals**

Case approvals are decision points at which one or more users decide whether to approve or reject a case.

**Channels**

A **channel** is a messaging service, voice service, web portal or mobile portal. Customers interact with organizations through a variety of channels.

**LOW CODE APPLICATION DEVELOPMENT**

****

Use the **Taxonomy** component to define the business structure for your organization.

Use the **Constraints** component to implement channel limits and constraints.

Use the **Engagement policy** component to define the rules that control which actions are offered to which customers.

Use the **Arbitration** component to configure how actions are prioritized.

Use the **Channels** component to configure when and where Next-Best-Action is triggered.