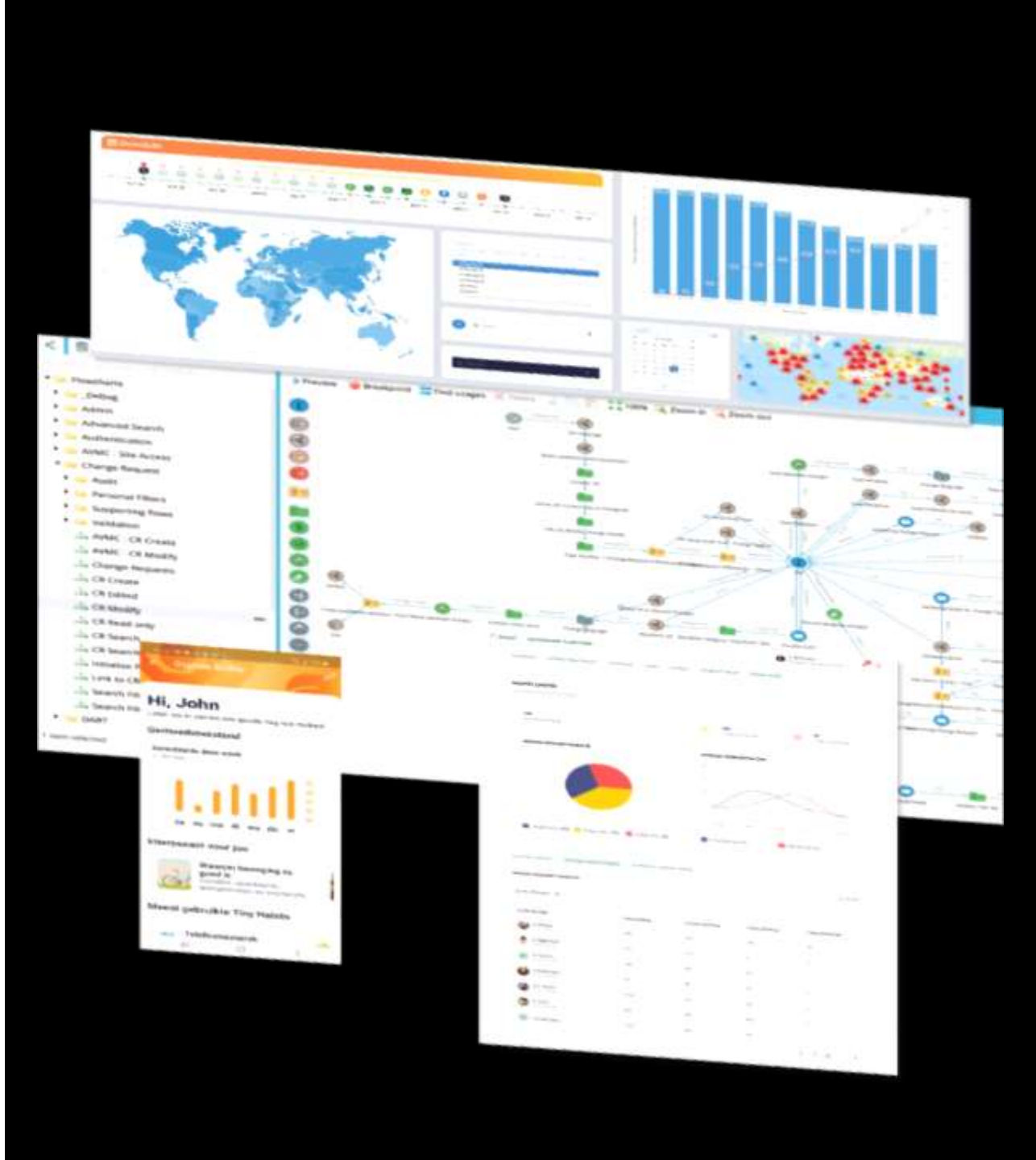
S-Square

TRUSTED . TESTED . COMMITTED

CERTIFIED NMSDC MBE 2018

S-Square - LowCode/NoCode (LC/NC) Enabling Technology Presentation



Jeff Friedman,
VP, Sales & Customer Success

Version - 20221215_V1

Current Challenges in Traditional Application Development

Long Development Timelines

- Custom development with standard SDLC processes
- Long incubation period before seeing a MVP
- Minor changes require long turn around time for design, build and testing.

High Capital Expenditure and Operating Costs

- Investment in Software platforms and Infrastructure for custom development
- Higher support costs due to diverse support requirements

Disparate Technology Landscape

- Multiple small projects using disparate technologies
- No uniform platform to manage small developments

Developer Shortages

- Developer shortages and skill-set challenges
- Multiple small productivity projects get deprioritized

6 Generations of Programming Languages

First generation (1GL) - machine-level programming language used to program first-generation computers
Examples: machine-level programming languages

Second generation (2GL) - assembly languages. Examples: Assembly

Third generation (3GL) - more machine-independent (portable) and more abstract therefore more programmer-friendly than previous generations of languages
Examples: Fortran, COBOL, BASIC, Pascal, C, C++, Perl, Python, Java, JavaScript, Ruby, PHP, C#

Fourth generation (4GL) - include support for database management, report generation, mathematical optimization, GUI development, or web development. Examples: ABAP, Unix Shell, SQL, PL/SQL, Oracle Reports, R

Fifth generation (5GL) - any programming language based on problem-solving using constraints given to the program to make the computer solve a given problem without the programmer, rather than using an algorithm written by a programmer. Examples: Prolog, OPS5, Mercury

Sixth generation (6GL) - programming language based on visual development. The overall umbrella term for these is "NoCode". Examples: Appian, WEM.io, Bubble.io

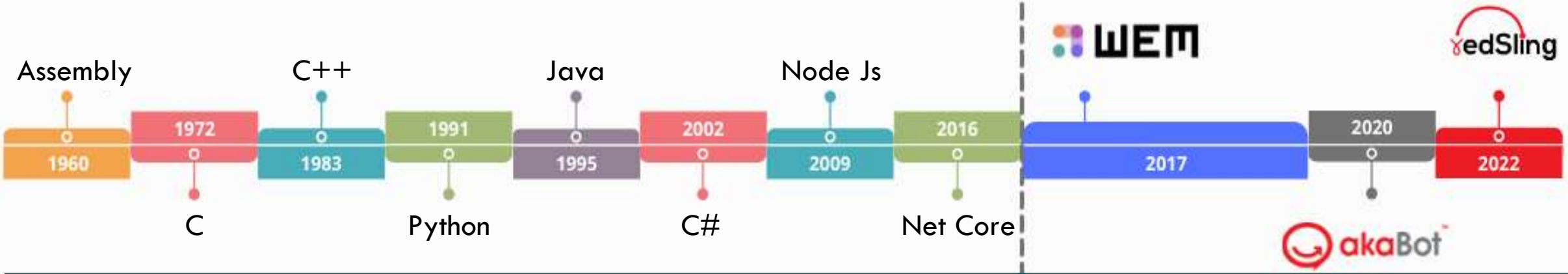
Reinventing Software Development

Traditional Coding

Requiring expensive, hard to retain code-linquists

No-Code

Empowering transforming support to employ business-knowledgeable techno-functional resources



Traditional computer languages require programmers to translate their thinking process into code built for the CPU and memory

Optimized for how we humans think. Converting natural thinking process into working software

Digital Transformation. Legacy Modernization. Business Velocity.

80%

COST REDUCTION

Empowers employing business knowledgeable (techno-functional) resources instead of costly, hard to retain code-linquists to build, deploy and maintain secure scalable enterprise-grade software.

10%

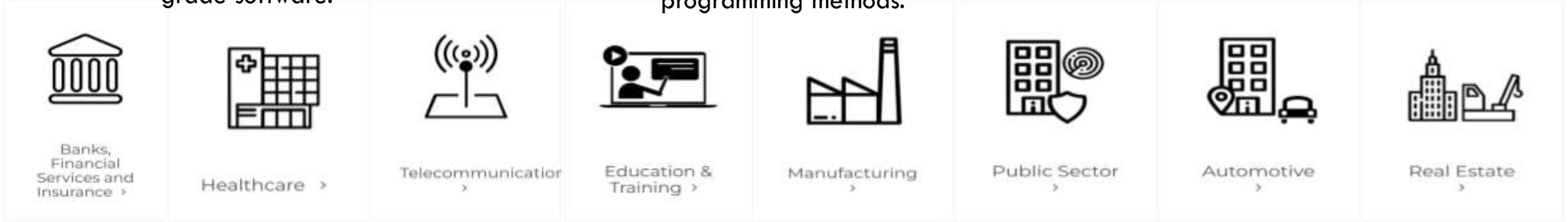
FASTER TIME-TO-MARKET

View app development in real-time. Deploy and update applications with a single click. Deliver software 10 times faster than traditional programming methods.

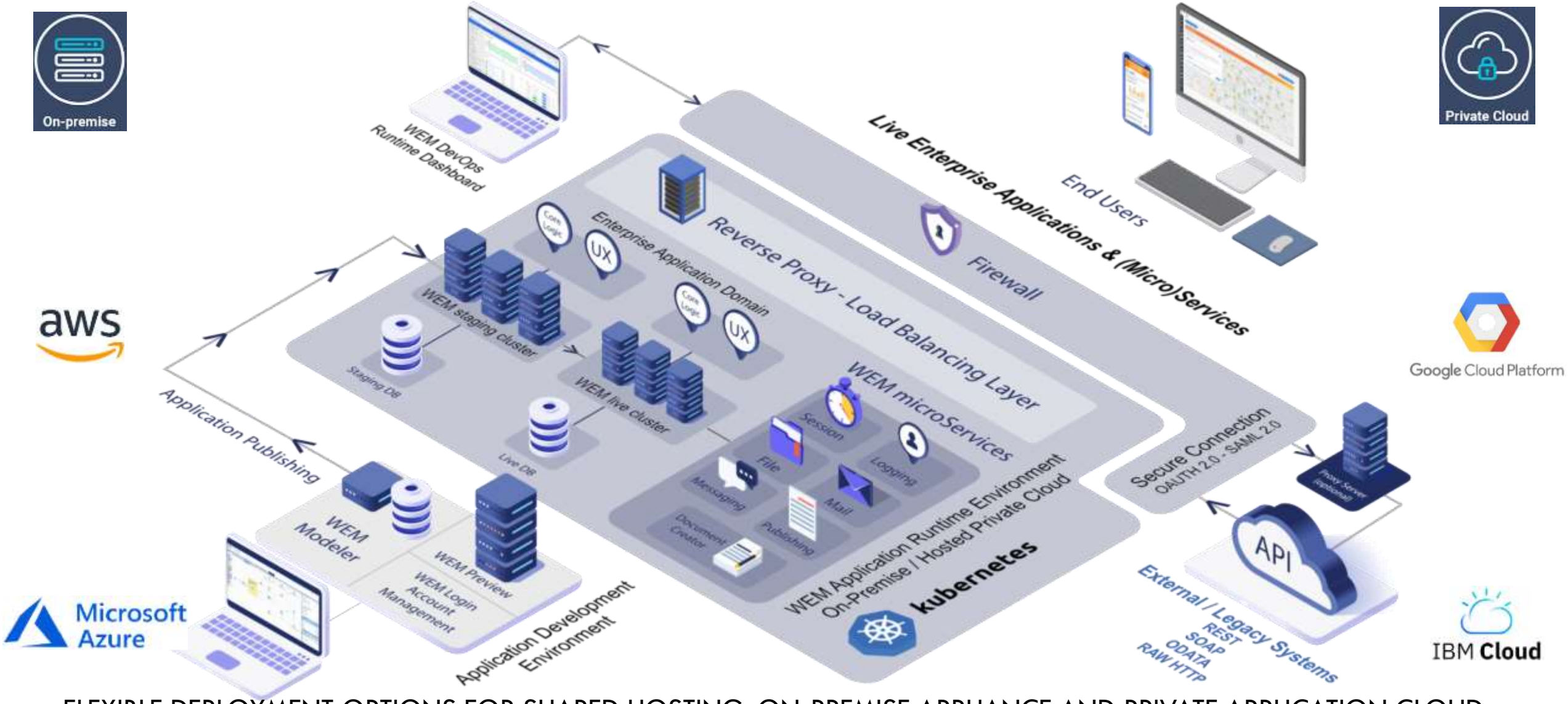
100%

ALIGNED TO BUSINESS

Translate innovative business ideas to custom software built with no code app builder at the speed of, and fully aligned with, business requirements.



SCALABLE, SECURE CLOUD ARCHITECTURE



FLEXIBLE DEPLOYMENT OPTIONS FOR SHARED HOSTING, ON-PREMISE APPLIANCE AND PRIVATE APPLICATION CLOUD

3rd Party LCNC Marketplace Product Evaluation

Criteria	WEM	Betty Blocks	Power Apps	OutSystems	Mendix
Category	No Code	Low code	Low code	Medium to high code	Low code
Platforms	Web, native apps	Web apps	Web, native apps	Web, native apps	Web, native apps
Data Model	Drag & Drop	Visual Editor	Tables	Visual Editor	Visual editor
Visual Editor	Web-based	For backend apps	Web-based	Many designer	Web-based, desktop-based
Workflows	Drag & Drop	Action Modeler	MS Flow	Visual modeler	Visual modeler
Look & Feel	Custom templates	Custom js/css/html	Customizable	Custom js/css	Custom js/css
Environment	Public, private cloud, on premise	Public cloud, on premise	Public, private cloud, on premise	Public, private cloud, on premise	Public, private cloud, on premise
Release Management	Fully	Fully	Partially	Fully	Fully
Integration	All API standards	JSON, SOAP/REST	Office365, REST	SOAP/REST	SOAP/REST

Use Case – Digital Banking

The primary objective of this cooperative banking society is to help distressed families with funds for temporary accommodation, repayment of the debt, medical emergencies, etc.

CUSTOMER CHALLENGES

- The necessity of Neo Banking and Open Banking API to get the Account data for the clients.
- They required the client's perspective about endorsing or rejecting the check dependent on the accuracy of the information.
- The user should be able to store data and the cheque, utilizing a portable application with the help of the camera; the client needed to tap the photos of the front and closing page of the check and present the structure.
- There was no such dashboard present to show all accounts including cheques deposited recently and other activities.

WEM ADVANTAGES

- **Use of existing data from legacy systems/integration with legacy systems**
- **Cloud solution offers flexible workspaces (not tied to a location)**
- **Easy to extend the application**
- **Fast return on investment.**

PROBLEM

Keeping tradition intact and embracing technology, the Bank has matched strides with India's digital revolution by offering various digital banking services. They needed customer-friendly access towards digitization such as dashboards, mobile apps, new technology evolved solutions.

SOLUTION

Open Banking API was used to get the account information for the users.

A dashboard was developed, to show all of the user's accounts including cheques deposited recently. Customers may approve or reject the cheque based on the correctness of the data. The customer can deposit the cheque using the mobile app, wherein the customer needs to click the pictures of the front and back page of the cheque and submit the form. A 100% digital process for cheque deposits and encashment was developed. The solution so developed is a one-stop-shop to get all the information of the inward and outward cheque transactions in a single dashboard eliminating the need to visit the bank.

Representative WEM Enterprise Customers

Thank You

Jeff Friedman,
VP, Sales & Customer Success

S-Square Systems, Inc.
4225 Executive Square Suite 600
La Jolla, CA 92037
+1 858-213-7063, +1 858-764-4441

S-Square
TRUSTED . TESTED . COMMITTED