LEGACY SYSTEM MODERNIZATION

Challenges

What are the main problem(s) your company, service, or product is trying to solve?



Impact on productivity and efficiency

Identification of key responsibilities

and infrastructure

Resistance to change from stakeholders and employees

Potential for service disruptions and downtime

Training and upskilling programs for employees

Enabling data-driven

decision-making and

analytics

Flexibility to adapt to

evolving business

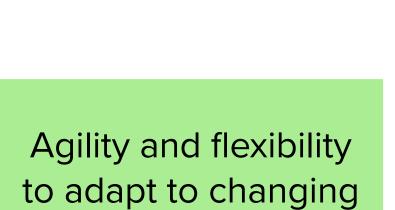
requirements and

market dynamics

Timeline and milestones for project execution

Aspirations

What are you trying to achieve? What does success look like for your team, project, or company?



business needs

Providing seamless

access across

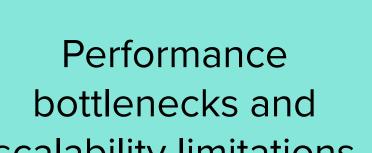
devices and channels

Data security and compliance enhancements

Embracing experimentation and learning from failures

Enhanced customer experience and satisfaction

What problems can you focus on that will have the most impact? What will you include or leave out of your strategy?



Implementing APIdriven and eventdriven architectures for scalability limitations flexibility and agility

Data cleansing, transformation, and validation

Security testing and vulnerability

assessment

Focus Areas



Design thinking and user-centered design principles

Vendor support,

community adoption,

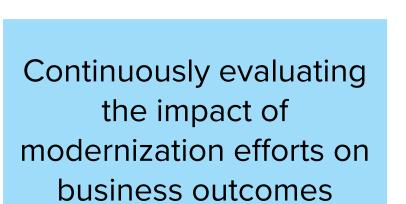
and ecosystem

maturity

API development and

management

What are the core values and principles that



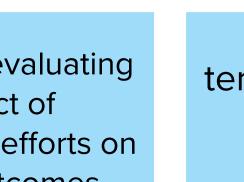
Decomposing monolithic systems into microservices or

modular components

Real-time monitoring and reporting capabilities

Guiding Principles

guide your team and help you make decisions?

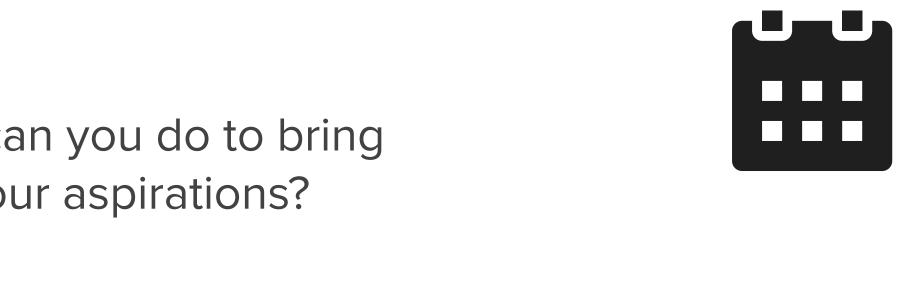


terative development and incremental updates

Leveraging middleware and ntegration platforms for connectivity

Activities

What kinds of things can you do to bring your team closer to your aspirations?



Inventory of existing systems, technologies, and dependencies

Functional requirements for new features and capabilities

requirements for new features and capabilities

Functional

Creation of architectural diagrams, design patterns, and guidelines

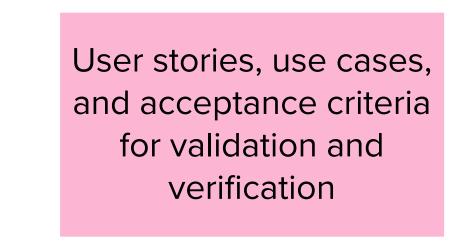
Development of new features, modules, or components

Unit testing, integration testing, and system testing

User training, onboarding, and support

Outcomes

Specifically, what will you measure to know you are successful? How can you be sure those metrics reflect success? Remember, metrics can be qualitative or quantitative.



Prioritized list of modernization initiatives and projects

detailing technical specifications, patterns, and guidelines

Design documents

Empathizing with end

users and

understanding their

needs

Encrypting sensitive

data in transit and at

Database schema and data models for data migration and management

Performance benchmarks, load test results, and security assessment findings

Environment setup and configuration documentation for infrastructure provisioning

FAQs, troubleshooting tips, and support contact information

