Proposed Team

PRODUCT TEAM

UX Team
- UX Lead Architect
  - UX Senior Engineer
  - UX Senior Engineer

DEV TEAM

BACKEND
- FSD-Api- Lead Architect
  - FSD-Api Senior Engineer
  - FSD-Api Senior Engineer

FRONTEND
- FSD-UI- Lead Architect
  - FSD-UI Senior Engineer
  - FSD-UI Senior Engineer

TEST TEAM
- Test Engineer
- Test Engineer

DEVSECOPS
- DevSecOps Engineer

Pricing

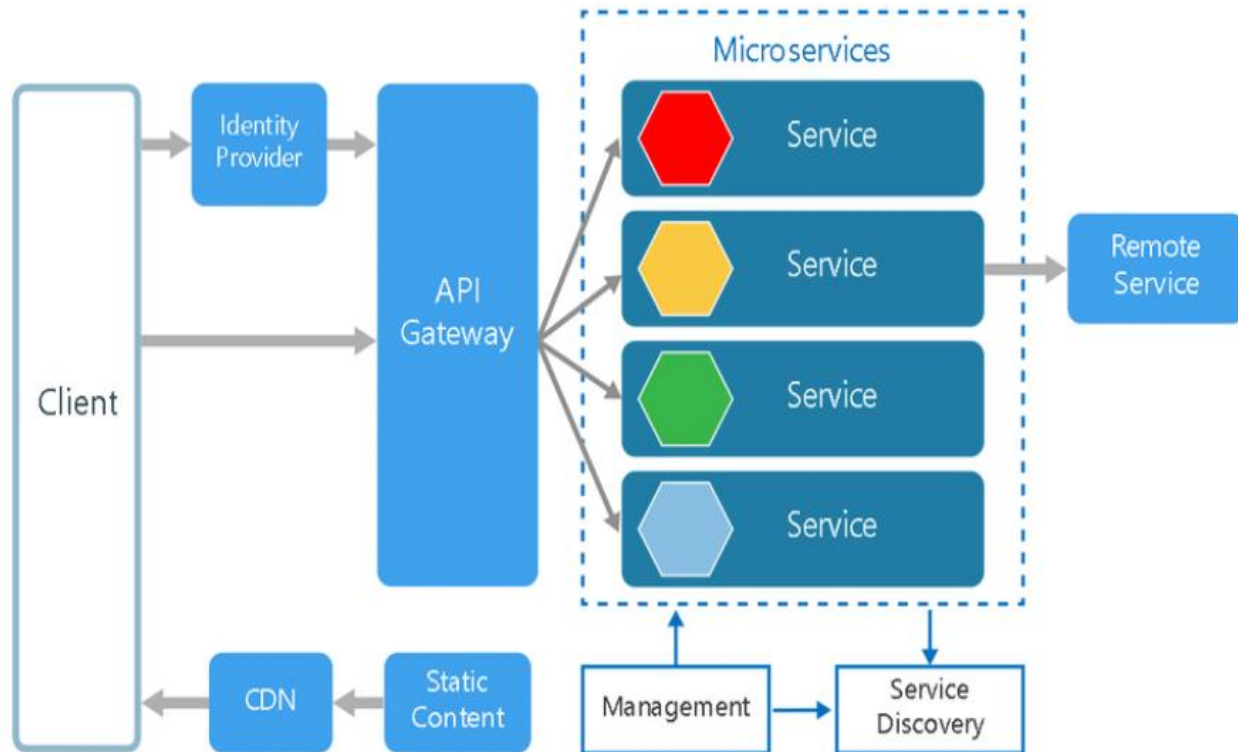| Role	Resources	Experience	Responsibility	Hourly	Monthly
FSD-Backend Api – Lead Architect	1	18 Yrs.	Ensure Enterprise SOA architectures, write SDKs, Framework	40$/Hr.	6000$/Month 6000*1=6000$
FSD-Backend Api – Senior Developer	2	5 Yrs.	Work primarily on SDK's and frameworks	20$/Hr.	3000$/Month 3000*2=6000$
FSD-Frontend UI – Lead Architect	1	10 Yrs.	Write UI libraries and components	30$/Hr.	4500$/Month 4500*1=4500$
FSD-Frontend UI – Senior Developer	2	5 Yrs.	Write UI components using libraries & framework	15$/Hr.	2400$/Month 2400*2=4800$
UX Designer Lead	1	10 Yrs.	Does business analysis , Information architecture, Themes , Designs	30$/Hr.	4500$/Month 4500*1=4500$
UX Designer	2	6 Yrs.	Does business analysis and Designs	15$/Hr.	2400$/Month 2400*2=4800$
AWS DevOps Engineer	1	8 Yrs.	Provision/ Automate AWS cloud services, maintain Env's and automate CI/CD, operations	30$/Hr.	4500$/Month 4500*1=4500$
Test Engineer	2	2 Yrs.	Do functional/api/performance testing with cucumber/postman gatling/JMeter	10$/Hr.	1500$/Month 1500*2=3000$
Total					38100$/Month

Outcome Based	Budget for estimated outcome volume. Payment based on result / SLAs.
Managed Services	Budget for monthly fixed cost and x% of variance cost. Payment based on delivered units.
Transaction based model	Budget for a specific transaction volume band per agreed duration. Payment based on transaction volume executed in agreed duration
Fixed Price (FP)	Specific / limited budget with predicted needs for few years. Payment based on Milestone/ project Completion.
Time & Material(T&M)	Floating budget. Payment based on Effort Put.
Hybrid (T&M for FP)	Phase wise budget, Phase1: Floating budget Phase2 : Specific / limited budget. Payment – Phase1 –Effort Based, Phase2 – Milestone / Project Completion.
Dedicated team	Fixed monthly budget. Payment based on resource.

While we are open to any flexible model, we strongly prefer, outcome based pricing model substantiated by our open product driven solutioning that protect customer investments.

Architectures & Development
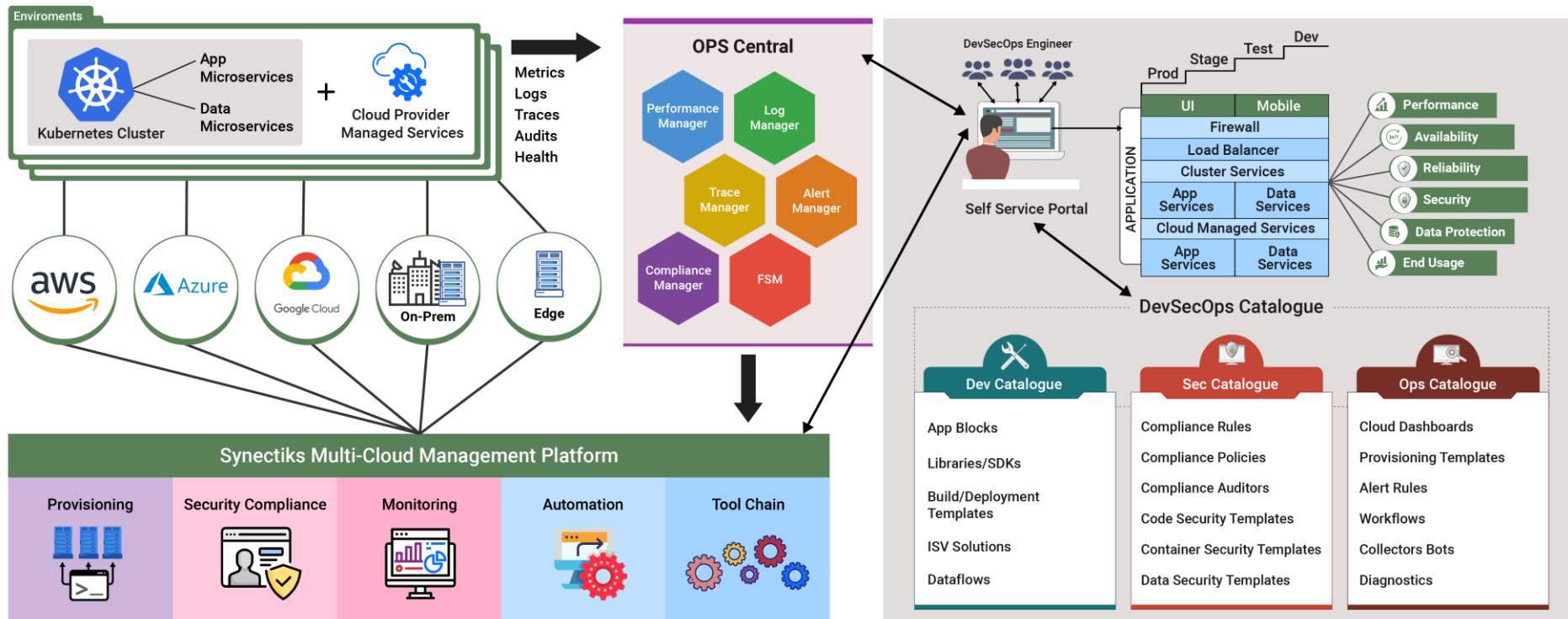
Development Principle

We build custom solutions based on Cloud Native principles with micro-service-based architecture , driven by DevSecOps approach to add development velocity and site reliability. While we are willing to be flexible on our execution approach, we believe that adopting the most modern technology stacks and architecture principles are critical to the project success.

Operation Model

Enviroments

Kubernetes Cluster
- App Microservices
- Data Microservices

+ Cloud Provider Managed Services

Metrics
Logs
Traces
Audits
Health

aws | Azure | Google Cloud | On-Prem | Edge

Synectiks Multi-Cloud Management Platform

| Provisioning | Security Compliance | Monitoring | Automation | Tool Chain |

OPS Central

- Performance Manager
- Log Manager
- Trace Manager
- Alert Manager
- Compliance Manager
- FSM

DevSecOps Engineer

Self Service Portal

Prod | Stage | Test | Dev

APPLICATION

UI	Mobile
Firewall	
Load Balancer	
Cluster Services	
App Services	Data Services
Cloud Managed Services	
App Services	Data Services

- Performance
- Availability
- Reliability
- Security
- Data Protection
- End Usage

DevSecOps Catalogue

Dev Catalogue

- App Blocks
- Libraries/SDKs
- Build/Deployment Templates
- ISV Solutions
- Dataflows

Sec Catalogue

- Compliance Rules
- Compliance Policies
- Compliance Auditors
- Code Security Templates
- Container Security Templates
- Data Security Templates

Ops Catalogue

- Cloud Dashboards
- Provisioning Templates
- Alert Rules
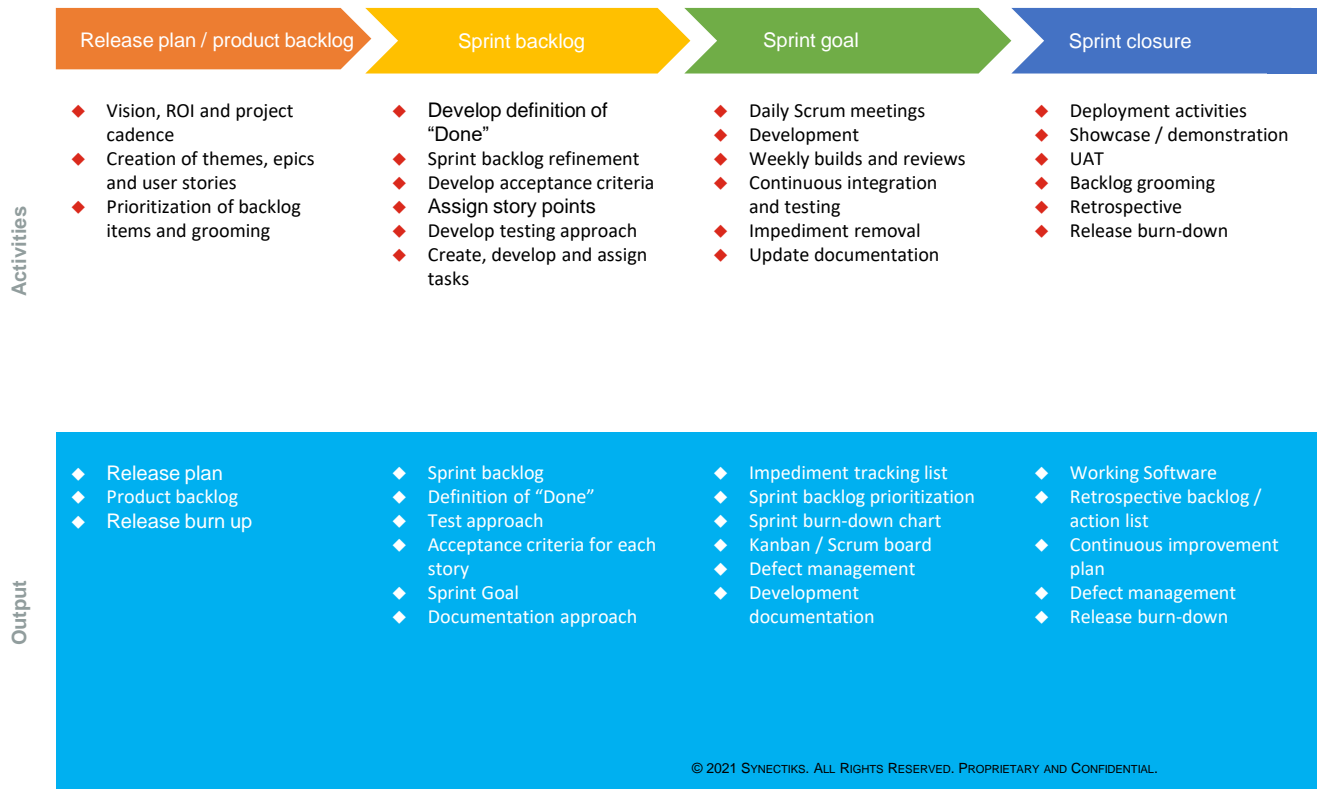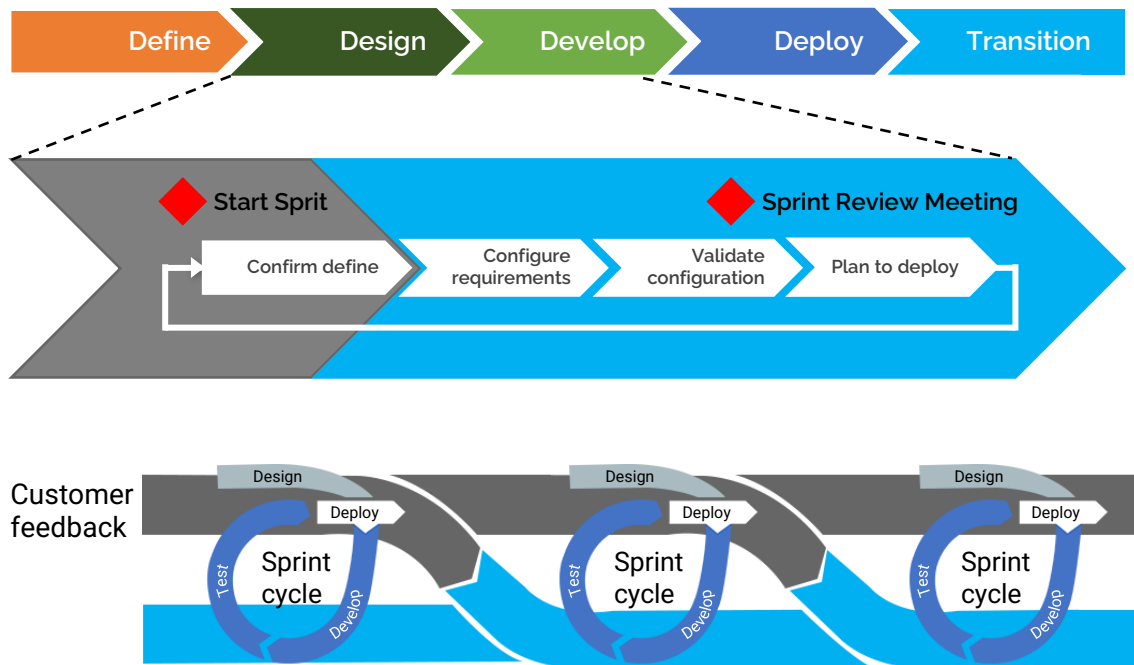- Workflows
- Collectors Bots
- Diagnostics

We adopt DevSecOps principles to add development velocity and site reliability. While we are willing to be flexible on our execution approach, we believe that adopting the most modern technology stacks and architecture principles are critical to the project success.

Process & Approaches

Methodology

When recommending an Agile execution approach, Synectiks considers the following core activities and concepts to be the most critical to the success of project execution. While we are willing to be flexible on our execution approach, we believe that eliminating these activities reduces the effectiveness of the agile approach.

Activities

Release plan / product backlog	Sprint backlog	Sprint goal	Sprint closure
◆ Vision, ROI and project cadence ◆ Creation of themes, epics and user stories ◆ Prioritization of backlog items and grooming	◆ Develop definition of "Done" ◆ Sprint backlog refinement ◆ Develop acceptance criteria ◆ Assign story points ◆ Develop testing approach ◆ Create, develop and assign tasks	◆ Daily Scrum meetings ◆ Development ◆ Weekly builds and reviews ◆ Continuous integration and testing ◆ Impediment removal ◆ Update documentation	◆ Deployment activities ◆ Showcase / demonstration ◆ UAT ◆ Backlog grooming ◆ Retrospective ◆ Release burn-down

Output

◆ Release plan ◆ Product backlog ◆ Release burn up	◆ Sprint backlog ◆ Definition of "Done" ◆ Test approach ◆ Acceptance criteria for each story ◆ Sprint Goal ◆ Documentation approach	◆ Impediment tracking list ◆ Sprint backlog prioritization ◆ Sprint burn-down chart ◆ Kanban / Scrum board ◆ Defect management ◆ Development documentation	◆ Working Software ◆ Retrospective backlog / action list ◆ Continuous improvement plan ◆ Defect management ◆ Release burn-down

Assumptions:

- 2 weeks design / development sprints
- Daily scrum check-in meetings

Customer Roles & Responsibility
- Require Product Owner role from customer to Articulate product strategy, vision and roadmap to stakeholders

- Prioritize and manage product back log.

- Synectiks representative to participate in daily scrum, sprint planning and retrospect meetings.

Synectiks Roles & Responsibility
- Synectiks PM along with developers participate in daily scrum and report status
- Synectiks PM mitigate any risk from software delivery perspective.

Define | Design | Develop | Deploy | Transition

Start Sprint | Sprint Review Meeting

Confirm define | Configure requirements | Validate configuration | Plan to deploy

Customer feedback

Design | Deploy | Test | Sprint cycle | Develop

Design | Deploy | Test | Sprint cycle | Develop

Design | Deploy | Test | Sprint cycle | Develop

7

Timeline

Sample Project Timeline

M1 S1	M1 S2	M2 S3	M2 S4	M3 S5	M3 S6	M4 S7	M4 S8

Future Phases

Sprints

S0 S1   S2   S3   S4   S5   S6   S7   S8

Design Sprints

Production Movement Sprints

Development Sprints

Requirements Documentation

Implementation IT1 Modules

Implementation IT1 Modules

Prod Environment Setup

Acceptance

Use Case

Implementation

Implementation

Implementation

Verification

Deployment

Iteration 1

Iteration 2

Iteration 3

Activities
- Use Case
- UX Design
- Review

Artifacts
- Use Cases
- Design Prototype

Activities
- Implement PH1 Modules
- Deployment
- User Guide

Artifacts
- IT1 Modules
- User Guide

Activities
- Implement IT2 Modules
- Deployment
- User Guide

Artifacts
- IT2 Modules
- User Guide

Activities
- Implement IT3 Modules
- Feed Back / Changes
- User Guide

Artifacts
- IT3 Modules
- User Guide

Activities
- Deployment
- Release
- Operation

Artifacts
- Production App

Activities
- Test / Change

Artifacts
- ATP results