Maturity Levels Components		0	1	2	3	4	5	6	7	8	9	10
1.	**Lifecycle Management**											
1.1.	Version Management			Implement Evolutionary API Strategy			Implement Multiple API Versions Strategy	Implement API Deprecation Protocol	Check Backwards Compatibility			
1.2.	Decoupling API & Application		Decouple API & Software Versioning			Decouple Internal & External Data Model	Decouple Internal & External Data Format	Decouple Internal & External Transport Protocol				
1.3.	Update Notification			Distribute Changelogs	Distribute Versioning Notification through Channel(s)		Extend API with Versioning Information				Announce API Versioning Roadmap	
2.	**Security**											
2.1.	Authentication		Implement Basic Authentication			Implement Authentication Protocol			Implement Single Sign-on			
2.2.	Authorization			Implement Access Control		Implement Token Management		Implement Standardized Authorization Protocol	Implement Authorization Scopes			
2.3.	Threat Detection & Protection		Implement Allow & Deny IP Address Lists	Implement Injection Threat Protection Policies			Implement DoS Protection		Implement Security Breach Protocol		Conduct Security Review	Implement Zero Trust Network Access
2.4.	Encryption		Implement Transport Layer Encryption		Implement Certificate Management							
3.	**Performance**											
3.1.	Resource management			Implement Load Balancing			Implement Scaling	Implement Failover Policies				Implement Predictive Scaling
3.2.	Traffic Management		Set Timeout Policies	Implement Request Caching	Perform Request Rate Limiting	Perform Request Rate Throttling	Manage Quota	Apply Data Volume Limits			Prioritize Traffic	
4.	**Observability**											
4.1.	Monitoring		Monitor API Health		Monitor API Performance		Monitor Resource Usage					
4.2.	Logging		Log Errors	Log Access Attempts	Log Activity		Audit User Activity					
4.3	Analytics			Report Errors	Broadcast API Status			Generate Custom Analysis Reports	Set Alerts		Enable Predictive Analytics	
5.	**Community**											
5.1.	Developer Onboarding		Facilitate Developer Registration			Provide SDK Support	Implement Interactive API Console			Provide Sandbox Environment		
5.2.	Support		Establish Communication Channel			Manage Support Issues		Dedicate Developer Support Team				
5.3.	Documentation		Use Standard for Reference Documentation		Provide Start-up Documentation Including Samples		Create Video Tutorials					
5.4.	Community Engagement		Maintain Social Media Presence		Provide Community Forum	Provide Developer Portal			Organize Events		Dedicate API Evangelist	
5.5.	Portfolio Management		Enable API Discovery			Provide API Catalog	Bundle APIs					
6.	**Commercial**											
6.1.	Service-Level Agreements		Publish Informal SLA		Provide SLA			Monitor SLA Proactively	Customize Personalized SLA			
6.2.	Monetization Strategy							Adopt Subscription-Based Monetization Model		Adopt Tier-Based Monetization Model	Adopt Freemium Monetization Model	Adopt Metering-Based Monetization Model
6.3..	Account Management			Implement Subscription Management System					Report on API Program Business Value	Provide Subscription Report to Customer	Proactively Suggest Optimizations to Customer	