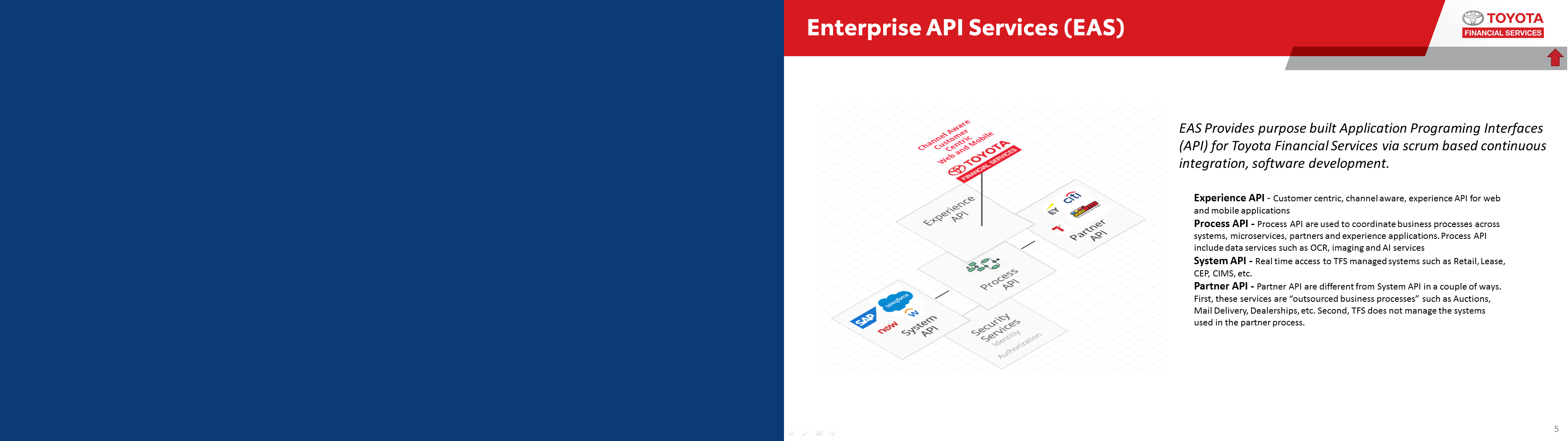
**Enterprise API Services**



|  |  |
| --- | --- |
| **Organization Name:** | Enterprise API Services (EAS) |
| **Description of Products:** | EAS Provides purpose built Application Programing Interfaces (API) for Toyota Financial Services via scrum based continuous integration, software development.  **Experience API** – Customer centric, channel aware, experience API for web and mobile applications  **System API** – Real time access to TFS managed systems such as Retail, Lease, CEP, CIMS, etc.  **Partner API** – Partner API are different from System API in a couple of ways. First, these services are “outsourced business processes” such as Auctions, Mail Delivery, Dealerships, etc. Second, TFS does not manage the systems used in the partner process.  **Process API** – Process API are used to coordinate business processes across systems, microservices, partners and experience applications. Process API include data services such as OCR, imaging and AI services  THE API catalogue includes a shared definition of Business Capability, API orchestration, Business APIs, Master Data Entities, Translation of Core Data attributes, and enterprise Events |
| **Roles:** | **EAS National Manager “**Factory Owner**” –** EAS National manager is responsible for the overall health of the EAS agile factory and ALL of its products and scrum teams. EAS national manager leads the change to agile; manages up and across TFS and Toyota affiliates; builds teams and defines the mission and vision; coaches new teams; removes obstacles for all. EAS national manager is a servant leader who enables all the factory teams to self-organize, self-manage, and deliver results via effective Lean-Agile practices.  **Product Owner(**s) - Product Owner(s) are responsible for managing expectations of customers, users, Platform Release Management (across EDP, EAS, etc.), and other stakeholders. They are responsible for Product Backlog Management including ensuring sufficient items are available for squads, grooming and prioritizing the backlog. Product manager’s work directly with Release Managers to decide what to deliver (release) to customers/users to meet enterprise planning needs  **Scrum Master**(s) - The Scrum Master removes impediments of all kinds and enables squads to self-organize, self-manage, and delivers results via effective Lean-Agile practices. The Scrum Master supports and enforces the Scrum process and other rules that the squad has agreed upon. The Scrum Master helps the squad coordinate with other squads and communicates status to management as needed. Scrum master follows the scrum process.  **Squad**(s) aka “Scrum Team” – All members of the squads are known as “Developer”. Developers play the traditional roles of analyst, quality assurance, development and deployment.  **Architect**(s) – Works with the product owner to realize the product vision. This is a key role which is always focused ahead on the architecture / design for next sprint, the “definition of done” for individual stories and to maintain overall system vision for a product owner.  **EAS Manager(**s) “Factory Manager” – EAS manager focuses on ensuring the EAS factory is running smoothly day to day. As such, this role focuses on the factories most important resource, people. EAS manger’s attention is on interviews, supervision, compensation, providing active coaching and assisting individual team members with career goals, training and advancement. The EAS manager leads and promotes agile practices and promotes Toyota Problem Solving (TBP) and Kaizen and keeps KPI and metrics for the team. The EAS managers handle all budgets, time off approval and work closely with Product owners to remove obstacles for the team. |
| **Ideal Situation (KPI):** | Each scrum team produces 60 or more “delivered story points” each sprint. |
| **Delivered story points:** | Delivered story points satisfy the definition of done which is developed, tested, integrated and documented per an agreement between the product owner and the scrum team. |
| **Target Organization Structure (for FY20):** | Two(2) Product Owners:   * **Experience and Partner API Product Owner** - One product owner for **Experience API** and **Partner API**. This product owner will focus on the API for web, mobile and partners * **System and Process API Product Owner -** One product owner for **System API** and **Process API**. This product owner will focus on the Retail, Lease, CIMS, and internal application and process API   Three(3) Squads each with four(4) developers (minimum):   * One(1) Squad focused on **Experience and Partner APIs** * Two(2) Squad focused on **System and Process APIs**   -  Three(3) Scrum Masters:   * One for each squad   Two(2) architects – shared by all  One(1) EAS Manager – shared by all  One(1) EAS National Manager – shared by all  One(1) Service Management Team – shared by all and provided as a service (the service management team is NOT part of the EAS factory) |
| **Sprints:** | Sprints will occur each month for each squad  Each squad should be capable of handling approximately (80) story points total of completed work for one (1) sprint  The actual number of story points in each sprint may vary.  At the end of each sprint, the actual number of delivered story points will be determined based on work accepted by Product Owner |
| **Sprint Planning Process:** | * Product owners and architects work with customers, users, Platform Release Management (across EDP, EAS, etc.), and other stakeholders to frame up a speculative release plan for the year. At this level, work is defined as epic level stories without detail. This release plan is is constantly evolving based on the business needs * As part of their daily activities, product owners and architects review the enterprise release plan (at least Three (3) months out) and create detailed user stories to support enterprise epics. These stories are formally entered into the Jira project for EAS. * At the beginning of each sprint, a Sprint Review meeting is conducted by Scrum Master with the squad and product owner and architect to create the sprint backlog * Each Week product owners, architects and squads conduct a one (1) hour story grooming session. This is a time boxed meeting to focus on the “definition of done” for the next sprint |
| **Sprint Estimation:** | For purposes of estimation:   * Each sprint will be 4 business weeks in length for a total of 20 business days each month * Each business day is eight (8) hours per person. |
| **User Story** | User stories document the requirements for the work to be accomplished in each sprint.  To start a sprint, the backlog of ready for sprint **estimated story points** should sum to sixty (60) to eighty (80) story points. **Product Owners** is responsible for ensuring that a backlog of user stories is “**ready for sprint**” at the start of each sprint.  **Ready for sprint** means a story has the following attributes written:   * Description, * Acceptance Criteria (“Definition of done”) * Estimated Story points   **Description** of the work to be accomplished sufficient to begin  **Acceptance Criteria** shall represent the “definition of done”. It is the responsibility of product owners and architects to define the definition of done **mutually with the squad** **prior** to completion of a user story.  **Estimated Story Points** are an estimate of the overall effort ***that will be required*** to fully implement a story.  When work for a user story is complete, Squad is responsible for recording the **Actual Story Points** for the user story and must designate **done** using the TFS Jira system.  **Actual Story Points** are an estimate of the overall effort ***that was required*** to fully implement the story. Actual story points will be mutually agreed upon by both the Squad and Product Owner prior to completion of a user story  When a user story is designated as “**done**” by the squad, this shall be taken as squad’s acceptance of the stories **acceptance criteria** and **actual story points.**  ALL Story attributes and state are “agile” and may change over the course of a sprint  Squads, architects, scrum masters and project managers and service management teams are empowered to suggest user stories to enhance the products and improve services. |
| **Story Points:** | Story points are a unit of measure for expressing an estimate of the overall effort that will be required to fully implement a story  For purposes of estimation:   * 1 story point shall represents 1 person’s (skilled in the “art” and/or role) work for one day (1 person/1 day = 1 story point). * 1 person shall plan to spend 1.5 hours (meetings) and 6.5 hours (work) each day |
| **Story Estimation:** | For purposes of estimation:  User Story  Extra Small 1 story point  Small 2 story point  Medium 5 story point  Large 8 story point  Extra Large 10 story point  Epic Story  Small 1 sprint  Medium 2 sprint  Large 3 sprint  Story estimation will improve as teams become more skilled. It is not expected that the story points for a task remain the same from one sprint to the next |
| **Backlog:** | TFS provided Jira system will be the master source of user stories  EAS will maintain its own backlog which will be fed from the common Platform Enablement backlog |
| **Scrum Ceremonies and roles all squads are expected to participate in:** | Scrum Master Directed meetings:   * Daily scrum (1/business day) * Sprint planning (1/sprint) * Sprint review (1/sprint) * Sprint retrospective (1/sprint)   Product Manager and Architect Directed Meetings:   * Story grooming (1/week) |
| **Scrum Team Key Capabilities:** | **Mulesoft (New), Axway and Tibco as API gateway, ESB and primary management plane**  **IBM Products**  IBM Cloud Private, zOS EE  **Core Technology Components**  Apache NiFi, Confluent Kafka, MongDB, PostgreSQL  **Infrastructure as a Code**  Terraform, Kubernetes, Docker, IBM Cloud Private, Amazon Web Services  **Repositories**  Jfrog Artifactory, GitHub Enterprise, ICP Helm, ICP Private Docker Image Repository  **Security Kanban**  SonarQube, ICP Vulnerability Advisor  **Architecture Patterns**  EIP Design and Delivered Frameworks (Prior Work), Microservice Development, Streaming Data Lake, API design, oAuth JWT  **Planning and Development**  Atlassian JIRA , Eclipse IDE, Spring (Java)  **Continuous Integration**  EIP CI Process (Prior Work) Jenkins, UrbanCode Deploy, Maven, Gradle, openJDK, Debian, Azule, apt-get, dockerhub,  **Functional Validation**  Junit, Cucumber, Selenium  **Control, Operations and Monitoring**  Kibana, Logstash, Grafana, ELK Stack, Istio, Ambassador  **Environments: AWS, TFS on premise, Heroku** |
| **Documentation of Design and Architecture, and Source Code** | Architects are expected to ensure software and systems is well documented and easy to understand using the following tools GitHub Enterprise, Swagger, Draw.io, Markdown.  All developers to act as “citizen developers” by self documenting code and including read me files written in Markdown on all repositories  Source code developed is to be maintained in TFS Github and is property of Toyota |

