Hewlett Packard Enterprise

Agile Delivery Model

Cal eStore

Notice on Export Controls: The technical data and technology inherent in this Document may be subject to the export, import, and other trade-related laws and regulations of the U.S. and other national governments. Reader agrees it will not export, import, re-export, or transfer in country, release or otherwise transfer by any means or method any items and/or technical data contained in this Document to countries or nationals of countries identified by the U.S. and other national governments as countries and nationals subject to trade control embargoes, sanctions, prohibitions and restrictions, including but not limited to all countries and nationals identified in country groups D:1, E:1 and E:2, as identified in Supplement No. 1 to Part 740 (http://www.access.gpo.gov/bis/ear/pdf/740spir.pdf) of the U.S. Export Administration Regulations, without first obtaining appropriate government authorizations.

Contents

Model Description

Introduction

Enterprise Services LLC (ES) has developed and fine-tuned the Global Agile Software Delivery Model to guide our project teams' delivery of projects using Agile delivery models. At its core our global Agile software delivery model is based on the Agile manifesto, principles and the scrum delivery model.

Scrum Framework

The Scrum framework is used throughout the applications industry for everything from simple projects to changing the way entire enterprises do their business. Scrum significantly increases productivity and reduces time to benefits while facilitating adaptive, lean systems development.

Based on modern empirical process control theory, Scrum supports a work process that optimizes the quality and level of functionality of products and solutions that can be created in a given period of time, given available resources. Useful product functionality can be delivered regularly as specifications, architecture, and design emerge and transform into working software, even when working with volatile requirements sets or technologies.

Scrum is an iterative, incremental framework for projects and product or application development. It structures development in cycles of work called sprints, which may range from one to four weeks in duration. The sprint length is consistent over the course of the project, and sprints follow one after the other without pause.

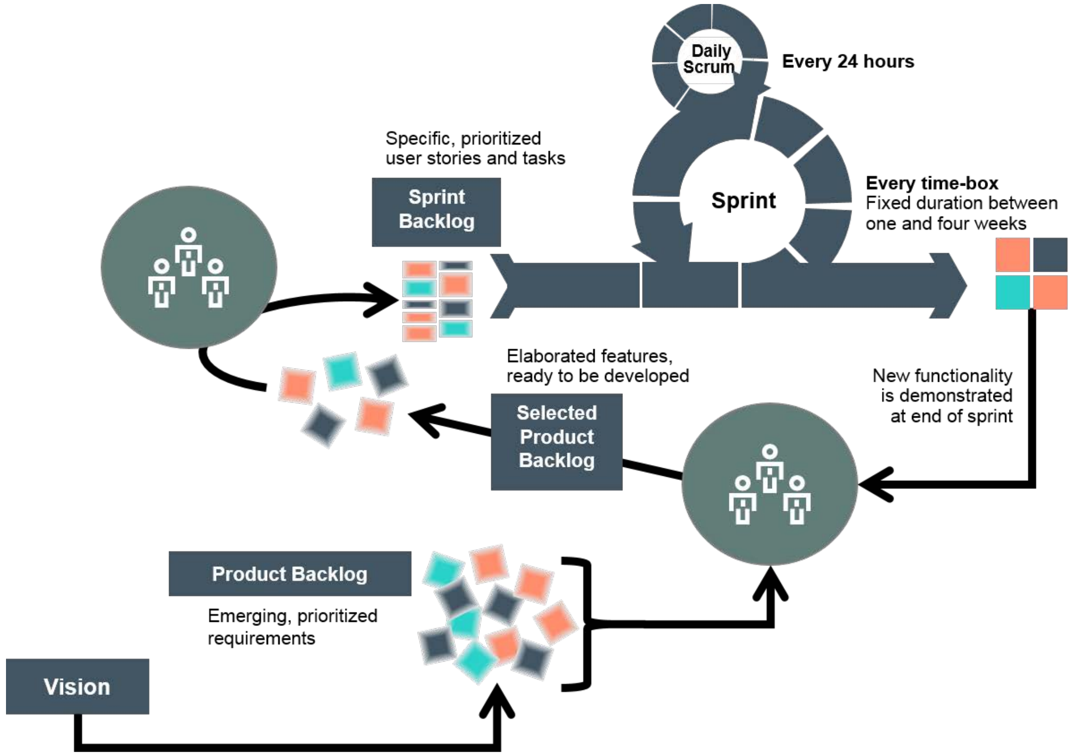
Work begins as the Product Manager articulates the product vision and puts together a product backlog that contains a set of high-level features to help realize that vision. At the beginning of each sprint, the cross-functional team selects items (specific features, user stories, customer or architectural requirements) from the product backlog, which has been prioritized under the direction of the Product Manager. The selected product backlog items are then elaborated such that they are ready to be fully estimated. The aim is to fill up the sprint backlog up to the known capacity of the team—the team velocity.

The team and the Product Manager agree on a mutually agreeable Definition of Done for product backlog items. For example, in the case of software, this means code that is integrated, fully tested, and potentially shippable, including any necessary installation, documentation, and training updates. Based on this Definition of Done, the team commits to fully complete the sprint backlog items by the end of the sprint.

During the sprint, the chosen items do not change. Every day the team gathers briefly to inspect its progress, and adjust the next steps needed to complete the remaining work. At the end of the sprint, the team reviews the sprint with stakeholders, and demonstrates what was built. Ongoing feedback identifies problem areas and relevant countermeasures are designed and incorporated into the next sprint, in a continuous improvement cycle.

The Scrum framework uses a set of interrelated roles, practices, and work products that optimize the flow of business value delivery, reduce organizational overhead and provide optimum adaptation to market demands through iterative delivery of product increments. The following figure provides a summary of the Scrum flow as used by Enterprise Services LLC.

Scrum Flow


Daily Scrum
Every 24 hours
Specific, prioritized user stories and tasks
Sprint Backlog
Sprint
Every time-box
Fixed duration between one and four weeks
Elaborated features, ready to be developed
Selected Product Backlog
New functionality is demonstrated at end of sprint
Product Backlog
Emerging, prioritized requirements
Vision

Although the Scrum framework is silent with respect to specific engineering practices, Enterprise Services (ES) expert practitioners enrich this process framework for optimum effectiveness within the project context with select techniques and processes.

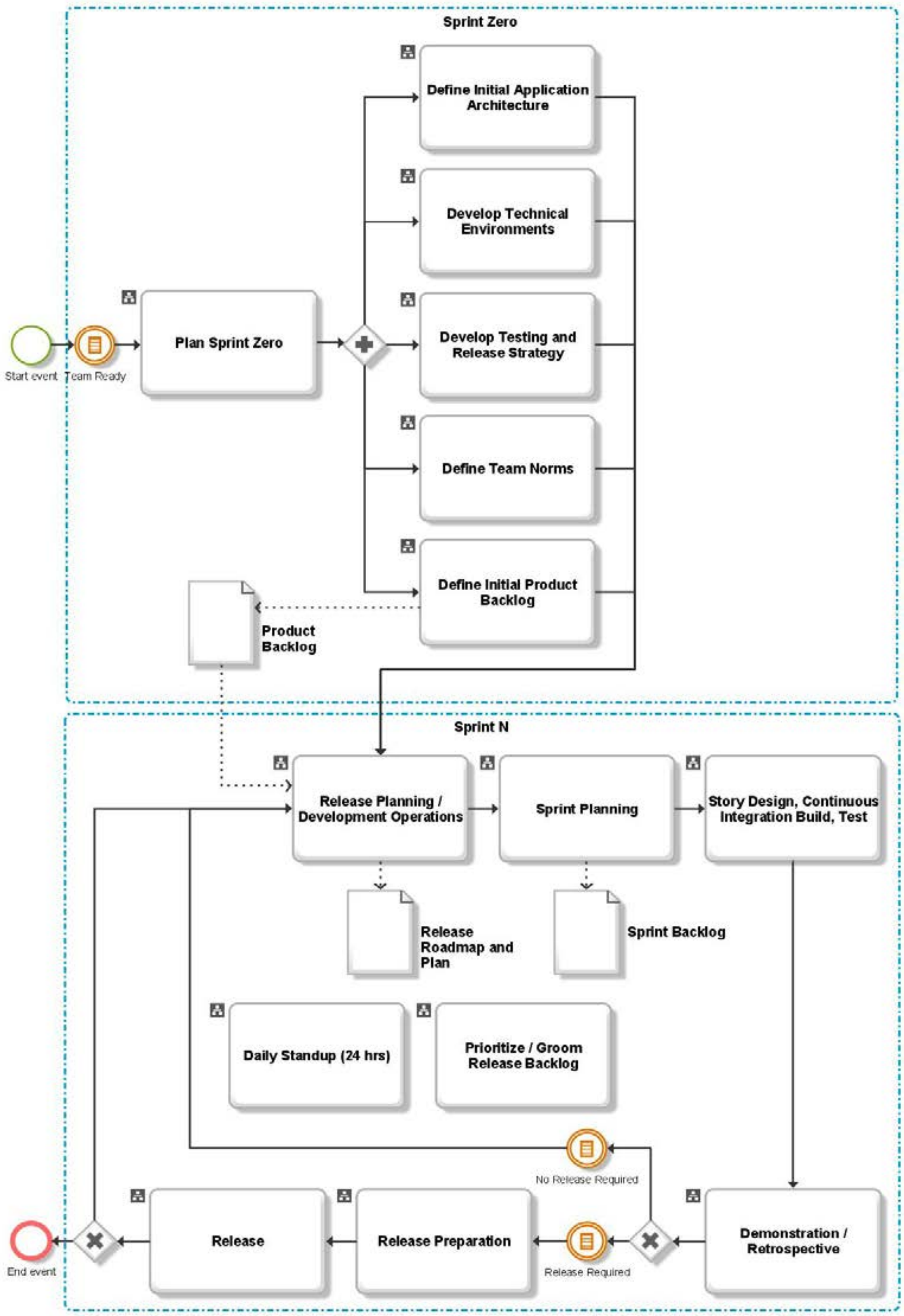
ES leverages the following extreme programming behaviors in all agile development:

- Simplicity — The team maximizes value creation.
- Communication — The team minimizes misunderstandings and error injection so the product achieves optimum quality.
- Feedback — The team focuses on adaptation and continuous improvement, so you get a product best fit for purpose.
- Respect — The Product Manager values the team’s expertise in technical matters, and the team values the Product Manager’s expertise in business matters. Responsibility is matched with authority, so the correct decisions are made for the right reasons.

- Courage — The team provides full transparency in terms of progress and estimates, so you can prioritize decisions.

The following diagram provides the overall process flow for the ES Agile development process.

Agile Process


Sprint Zero
Start event
Team Ready
Plan Sprint Zero
Define Initial Application Architecture
Develop Technical Environments
Develop Testing and Release Strategy
Define Team Norms
Define Initial Product Backlog
Product Backlog
Sprint N
Release Planning / Development Operations
Sprint Planning
Story Design, Continuous Integration Build, Test
Release Roadmap and Plan
Sprint Backlog
Daily Standup (24 hrs)
Prioritize / Groom Release Backlog
No Release Required
Release Required
End event
Release
Release Preparation
Demonstration / Retrospective

The following table describes each of the steps in our Agile process.

Agile Steps

Plan Sprint Zero	Validate the mechanisms, significant application components, techniques and tools necessary for the current release. Establish and validate the initial architecture of the solution, adapt the plan for the current release and upcoming iterations as required. Execute in one or more time-boxed iterations depending on the project needs.
Define Initial Application Architecture	Representatives from different capabilities (such as, Architecture, Data Engineering, Quality Assurance and others) work collaboratively to define high level approaches for development of the solution. Create the simplest architecture that can possibly work. The Architecture should be feature driven and sufficient for the upcoming release.
Develop Technical Environments	Initiate the establishment of the project technical environment to support the solution delivery. Establish technical environments to support engineering. Identify the need for acquiring, reallocating, modifying hardware and software components to support the project lifecycle, including testing needs. To review the technical infrastructure meeting the needs of the project and customer. Identify and start to plan for any changes needed to environments and tools, including the ability to communicate effectively between all locations via phone, email, instant messaging, shared storage, and team sites. This is to ensure excellent collaboration between people. Identify additional training needs and deliver planned training. Ensure the project non-functional requirements are addressed. Identify any client specified non-standard tools and initiate actions to ensure availability and access. Initiate waivers as needed for non-standard tools. Initiate facilities, systems management and licensing agreements as needed for the technical environment.
Develop Testing and Release Strategy	Provide guidance and establish a requirements-driven, risk-based framework within which all project testing activities will be defined, planned, executed and managed. Refer to the development standards to ensure alignment to the development approach. Select the release name(s) and theme(s), estimate initial team velocity, determine number of sprints in each release, and the sprint length.

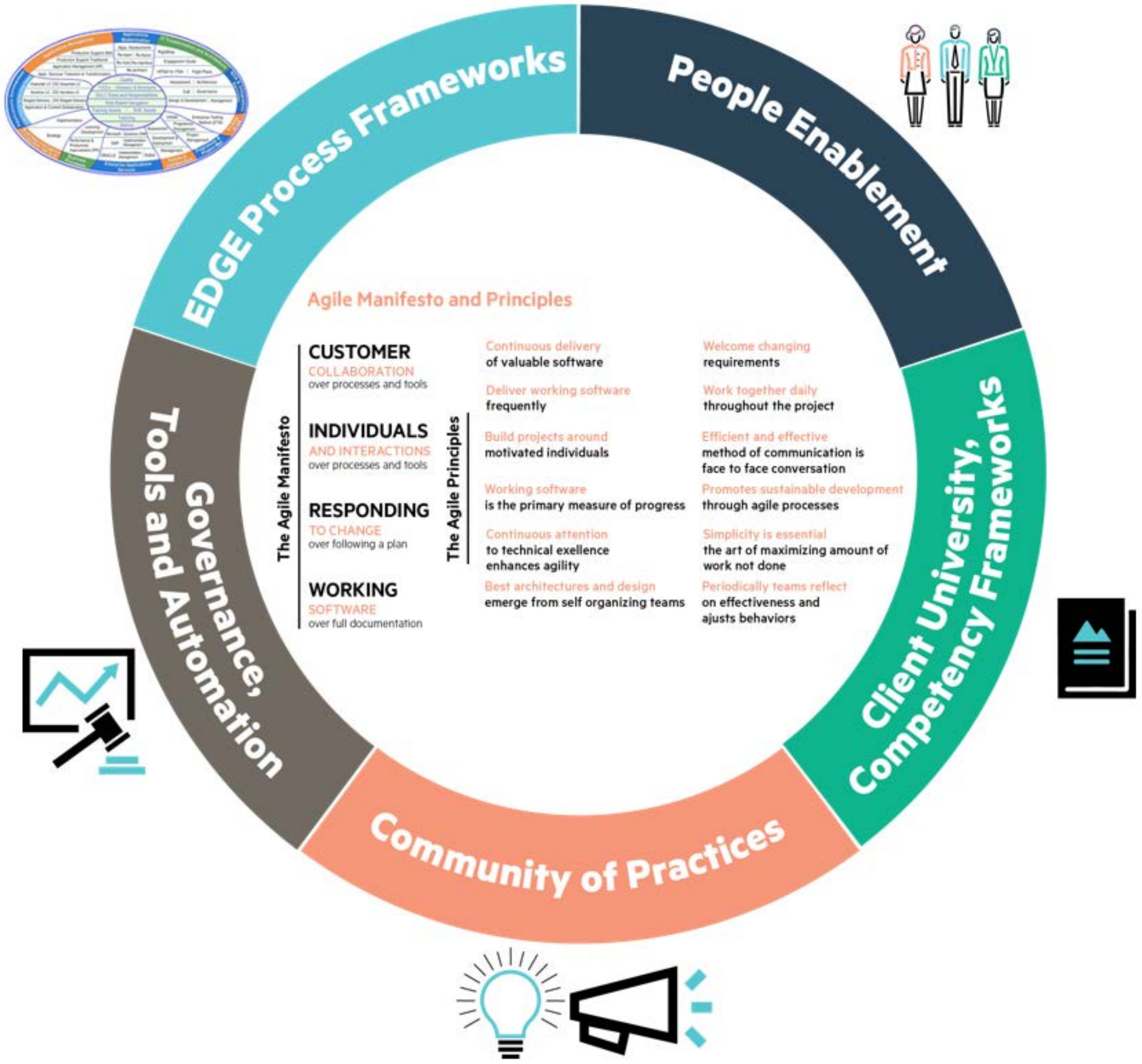
Define Team Norms	Review project charter, propose, project risks, and assumptions. Set up milestones, deliverables and timelines. Introduce stakeholders (their roles and responsibilities) and team members. Communicate the customer acceptance criteria and expectations on the applicable standards, regulatory compliance and security requirements. Identify the information that needs to be communicated to the team (internal/external), mechanism, frequency, content, responsibility, audience. Define the self-organizing standards and norms. Consider build frequency, naming conventions, tooling, index cards, task board, coding standards, timing of status meetings, team rules, test coverage, development techniques, and Definition of Done. Identify and document organizational or client process assets that apply and define how project-specific process assets needed will be provided. Plan the tools environment for the project execution. Determine the teams approach for conducting technical reviews. Work with the project technical leader to determine the triggers for conducting technical reviews on the project.
Define Initial Product Backlog	Produce the initial prioritized master list of all functionality desired in the product. At this stage, the backlog consists of very high-level, simple User Stories. The Product Backlog is a list of product features that satisfy a business need or a clear technical necessity such as a regulatory requirements, architectural constraints, back-end processing needed to make the application flow, and so forth. Features are prioritized according to business value, technical complexity, and associated risks and implications. The Product Backlog is continuously revisited by the Product Owner throughout project execution according to changing business demands, requirements and priorities. Newly introduced defects or any documentation-related work to be executed during subsequent sprints are included in the Product Backlog to be further prioritized.
Release Planning / Development Operations	Select the release name(s) and theme(s), estimate initial team velocity, determine number of sprints in each release, and the sprint length. The release plan contains release-level details of the project with a focus on guiding the execution of the current release. Various aspects of the plan may be kept at a high level and refined over time, or as soon as sufficient information becomes available.
Sprint Planning	Prepare a detailed work plan in support of an iteration called for by the project plan and to ensure that required resources (including staffing, training, budget, reusable assets, environments, and so forth) are available to begin performing the work.

Sprint Backlog	The Sprint Backlog is the set of Product Backlog items selected for the sprint, plus a plan for delivering the increment and realizing the Sprint Goal. It is a forecast by the Development Team about what functionality/service will be in the next increment and the work needed to deliver that functionality into a "Done" increment. The Sprint Backlog makes visible all of the work that the development identifies as necessary to meet the Sprint Goal. The Sprint Backlog is also a plan with enough detail that changes in progress can be understood in the Daily Standup Meeting. The Development Team modifies the Sprint Backlog throughout the sprint, and the Sprint Backlog emerges during the sprint. This emergence occurs as the Development Team works through the plan and learns more about the work needed to achieve the Sprint Goal.
Story Design, Continuous Integration Build, Test	Build the solution, using Agile techniques such as test-driven development, refactoring, and/or pair programming. For non-program related deliverables, develop the solution to satisfy the feature designs and user stories. Check in the application components that have passed unit testing and have good designs. Ensure that the checked-in code is integrated into an environment where the client and team have access for future integration and reviews. Work in the gaps that may exist between the development teams. Conduct unit testing according to documented testing specifications and the testing strategy. Track all problems discovered to resolution. Secure completed components and update documentation in accordance with configuration management plan.
Daily Standup (24 hrs)	A quick daily meeting of all members of the Scrum Team and the ScrumMaster. Track project status, monitor roadblocks and risks. Each person answers the following questions: • What have you done since the last meeting? • What will you do between now and the next meeting for this project? • What obstacles, if any, have you hit or can you see?
Prioritize / Groom Release Backlog	Continually decompose features into stories for the next sprints, particularly the next sprint to ensure that sprint planning can be executed without interruption.
Demonstration / Retrospective	Conduct a review/demo of the potentially shippable product increment as outlined in the quality plan. Demonstrate the accomplishments during this period by demonstrating new features. Assess the progress against the goals determined in the planning meeting. Ensure that the goal has been accomplished. Multiple reviews may be needed to ensure the demo is provided to internal and external (client) participants.

	Record actions identified during the review as new user stories in the Product Backlog to be prioritized for future sprints. Following the review, the team meets for the retrospective. The retrospective is an opportunity for the team to discuss what's working and what's not working, and agrees on changes to try. One approach taken is that the whole team gathers and discusses what they'd like to: • Start doing • Stop doing • Continue doing For Agile projects, this activity is performed at the end of each sprint. For iterative, non-Agile projects, this activity is performed at the end of each iteration. For non-Agile, non-iterative, this activity can be performed as needed.
Release Preparation	Package the application or solution for delivery, perform formal acceptance testing, gain final approval, and make the package available for release.
Release	Install all components of the solution in the technical environment, test installed applications, train all users and support personnel, and provide any start-up support according to the internal agreement.

Agile Ecosystem

Our Agile Ecosystem focuses on **People** and enables them with appropriate use of **Process** and **Tools**, as demonstrated in the following figure.



HPE has adopted EDGE, our repository of best practices, tools and templates, and Industrialized Delivery systems to enable standardization across our delivery centers and engagements while providing a flexible delivery model to suit our customer's business requirements. Though the Agile manifesto stresses individuals and interactions over process and tools, over time we have realized that process ensures consistency and repeatability which is essential for delivering at scale. Our observations have shown that technical competency, when supported with soft skills and domain knowledge, enhances employee productivity to helps us deliver value to our customers. We create dedicated universities focusing on domain, soft skills, process, and leadership skills to enable our teams to deliver high value to our customers. Community of Practices is an essential part of the ecosystem that enables us to propagate/reuse innovation and best practices across various customers. Governance ensures that we identify the right project, make the right reasonable commitments, and we deliver the right products to our customers.

Applying the Principles

Practices and the Team

For the Cal eStore our team adopted Scrum with XP practices as the delivery model for development. For the Cal eStore prototype, our team adopted Scrum with aspects of XP as the delivery model for development. The XP practice of shorter sprint iterations as well as flexibility in changing features not yet started within a sprint suited the compressed timelines of this particular exercise. Individuals in specific roles working from various locations comprised the team. Additional details are provided in this section.

The team selected the combination of XP practices along with Scrum because of the specific nature of the project. Almost all of the 12 practices were a perfect match for the Cal eStore project. These includes user story planning, small version product development, simple design, adhereing to a set of standards defined by the customer, automated testing, pair programming and continuous integration. As the project required fast turn around time keeping quality at the higest level, the team felt the use of XP principles would add value to the work being done.

People in specific roles and in various locations comprised the team. Additional details are provided in this section.

Details

The cross-functional scrum team includes the roles identified in the following table.

Roles and Responsibilities

Role	Responsibility
Product Manager	Final authority representing the customer's interest in backlog prioritization and requirement Is readily available to the team, especially during the sprint planning and review meetings
Product Expert	Proxy Product Manager; mitigates risks of executing the scrum in a distributed model Works closely with the Product Manager and scrum teams during sprint planning and reviews
Technical Architect	Overall application design Makes technology, architecture and tool decisions along with the client’s enterprise architect

Scrum Master	Facilitator for the team and Product Manager Cultivates creativity and promotes empowerment within the development team Maintains current information about the team's progress up to date and share that information with all parties
Development Team	Cross-functional, self-organized team that includes members in the following roles: • Business Analyst • Visual Designer • Front End Web Developer • Backend Web Developer • DevOps Engineer • Quality Assurance Analyst • Writer/Content Designer Designs, builds, tests and runs the application
Agile Coach	Coaches and mentors the team in Agile best practices Accelerates the team’s continuous improvement

Sprint 0

Project Initiation

Upon receipt of the RFI, the scrum team began working with the Product Manager to initiate Sprint 0 with the goal of defining:

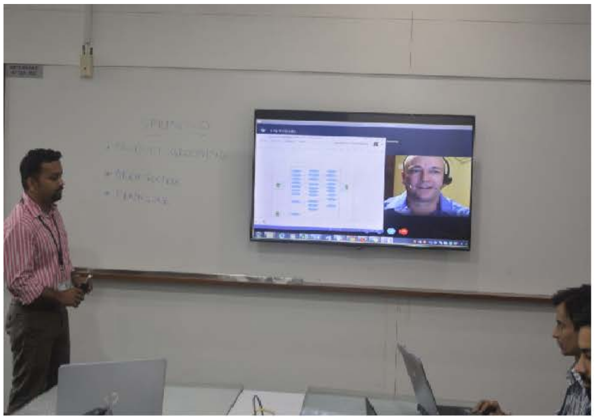
- The project vision
- Objectives
- Minimum Viable Product (MVP) (i.e., Epics and features in the Product Backlog)
- The release plan

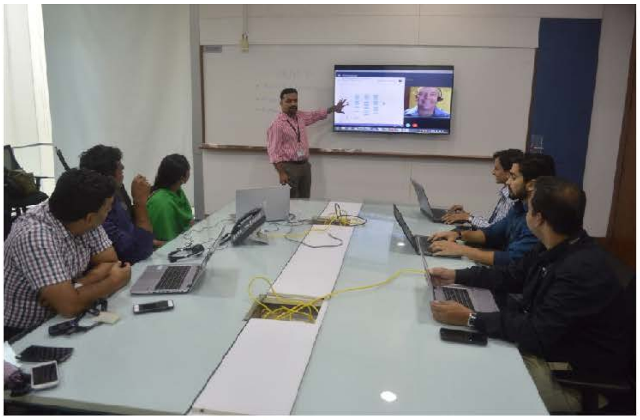
Workshop

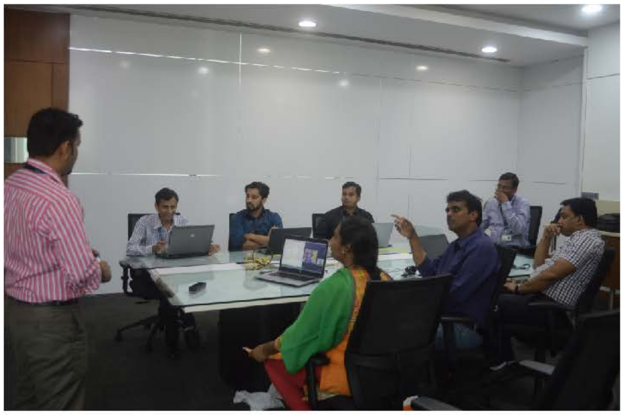
The vision for the project was elaborated into features by the Product Manager, product expert, solution architect, test lead, and Agile coach, as well as other team members during a workshop. The team defined the technical framework, test strategy, tools, development, test and higher environment strategy, access, etc. An epic wall was also defined. The sprint culminated with creation of a Product Backlog.

Sprint 0 Workshop

Photos in the following collection show team members at work during the Sprint 0 workshop.

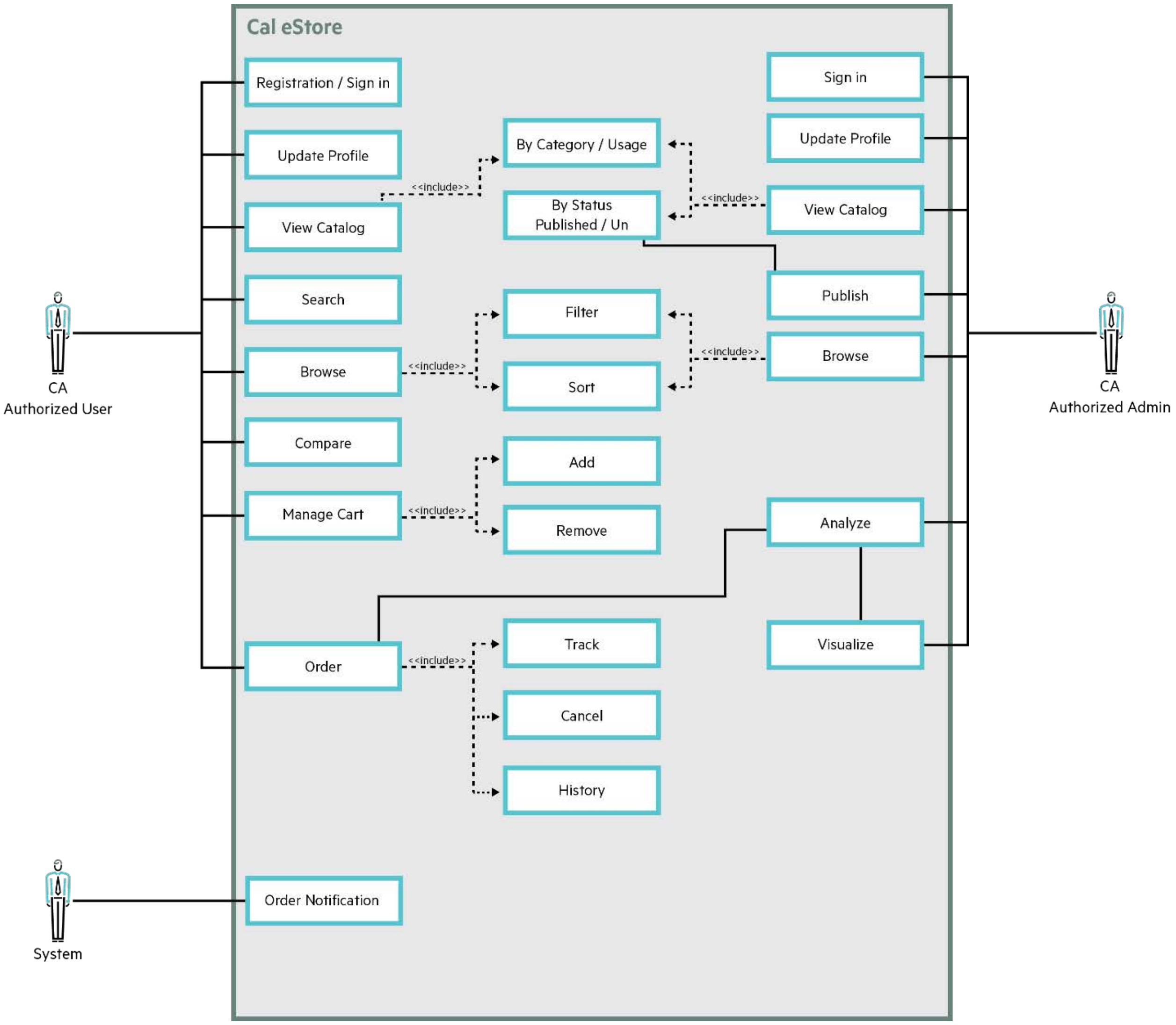






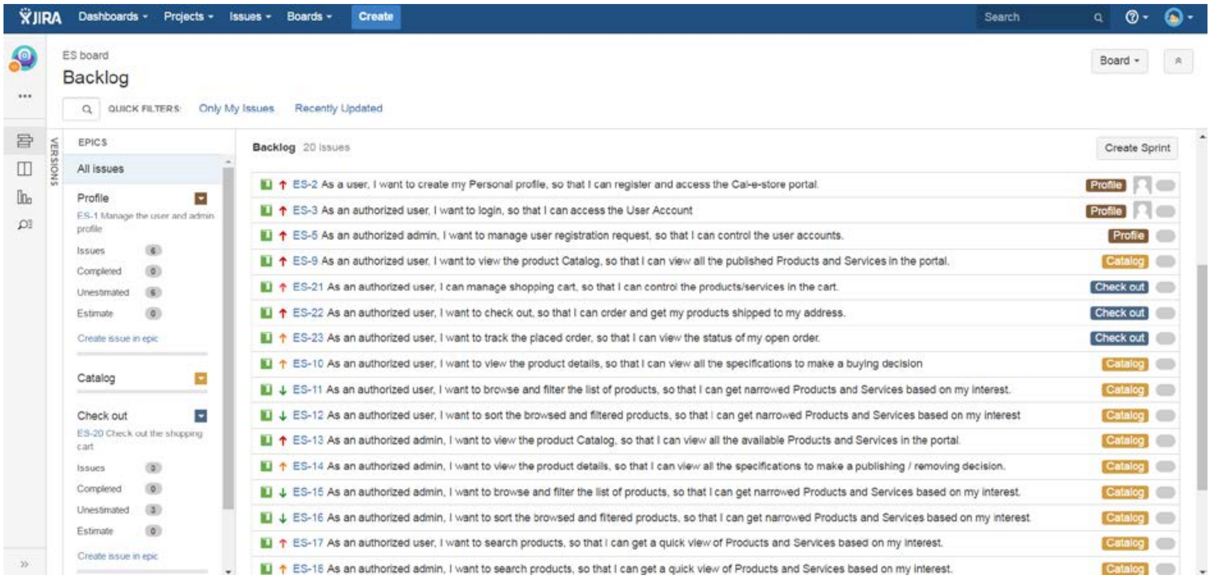
The following graphic depicts the Epic Wall for Cal eStore based on planning that occurred during Sprint 0.

Sprint 0 Output



The following screen capture shows the Product Backlog that was created as an output of Sprint 0. The Backlog was created in JIRA, an Agile Lifecycle management tool.

Sprint 0 Output



ES board

Backlog

EPICS

All issues

Profile
ES-1 Manage the user and admin profile
Issues 6
Completed 0
Unestimated 6
Estimate 0

Catalog

Check out
ES-20 Check out the shopping cart
Issues 3
Completed 0
Unestimated 3
Estimate 0

Backlog 20 issues

- ↑ ES-2 As a user, I want to create my Personal profile, so that I can register and access the Cal-e-store portal. Profile
- ↑ ES-3 As an authorized user, I want to login, so that I can access the User Account Profile
- ↑ ES-5 As an authorized admin, I want to manage user registration request, so that I can control the user accounts. Profile
- ↑ ES-9 As an authorized user, I want to view the product Catalog, so that I can view all the published Products and Services in the portal. Catalog
- ↑ ES-21 As an authorized user, I can manage shopping cart, so that I can control the products/services in the cart. Check out
- ↑ ES-22 As an authorized user, I want to check out, so that I can order and get my products shipped to my address. Check out
- ↑ ES-23 As an authorized user, I want to track the placed order, so that I can view the status of my open order. Check out
- ↑ ES-10 As an authorized user, I want to view the product details, so that I can view all the specifications to make a buying decision Catalog
- ↓ ES-11 As an authorized user, I want to browse and filter the list of products, so that I can get narrowed Products and Services based on my interest. Catalog
- ↓ ES-12 As an authorized user, I want to sort the browsed and filtered products, so that I can get narrowed Products and Services based on my interest Catalog
- ↑ ES-13 As an authorized admin, I want to view the product Catalog, so that I can view all the available Products and Services in the portal. Catalog
- ↑ ES-14 As an authorized admin, I want to view the product details, so that I can view all the specifications to make a publishing / removing decision. Catalog
- ↓ ES-15 As an authorized admin, I want to browse and filter the list of products, so that I can get narrowed Products and Services based on my interest. Catalog
- ↓ ES-16 As an authorized admin, I want to sort the browsed and filtered products, so that I can get narrowed Products and Services based on my interest Catalog
- ↑ ES-17 As an authorized user, I want to search products, so that I can get a quick view of Products and Services based on my interest. Catalog
- ↑ ES-18 As an authorized admin, I want to search products, so that I can get a quick view of Products and Services based on my interest. Catalog

Sprint 1

After Sprint 0, project teams interacted with business owners on a daily basis for continual elaboration of requirements and to refine the minimum viable product.

Sprint Planning

The team used time-boxed sprint cycles of one week. Each sprint cycle involved the additional elaboration of the requirements and development continually working toward our final product.

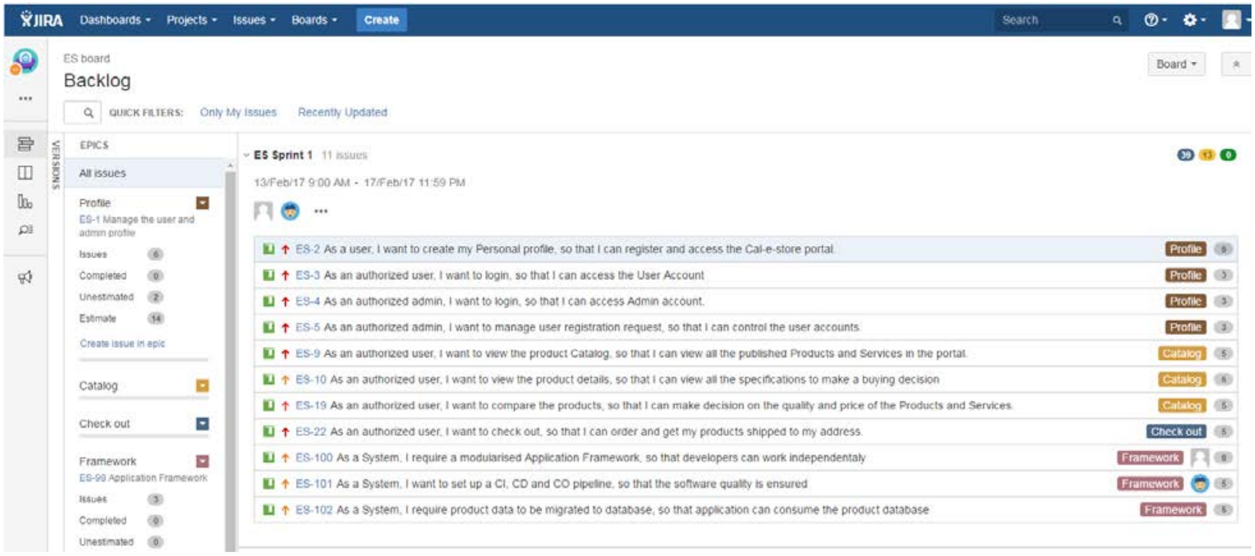
Identifying the Sprint Backlog

A sprint backlog was planned using the “planning poker method” and the session was limited to two hours. Planning poker is an Agile estimating and planning technique that is consensus based. The Product Manager describes a user story to the team. The team discusses the user story, questioning the Product Manager as needed. When the user story has been fully discussed, each developer estimates the size of the user story, usually called Story points. From this the team determines a value that becomes the estimate. The poker planning process may be repeated until consensus is achieved. Session participants included the Product Manager, product expert, scrum master and the development team. The following activities occurred during the session.

NOTE: The goal for Sprint 1 was to deliver a minimum viable application with basic end-to-end features. Additional features were moved to the subsequent sprints.

1. The Product Manager and product expert provided requirements with a priority for each.
2. The team discussed the sprint storyboard (i.e., candidates selected by the Product Manager on how the chosen work would be done.)
3. The team used the planning poker method to estimate the size of the sprint’s user stories. Once estimated, the group agreed upon the Definition of Done for the sprint.
4. The team committed to the features (user stories) to be delivered in the sprint and the Sprint Backlog was created.

Sprint Backlog Session



ES board

Backlog

All issues

Profile
ES-1 Manage the user and admin profile
Issues 6
Completed 0
Unestimated 2
Estimate 14
Create issue in epic

Catalog

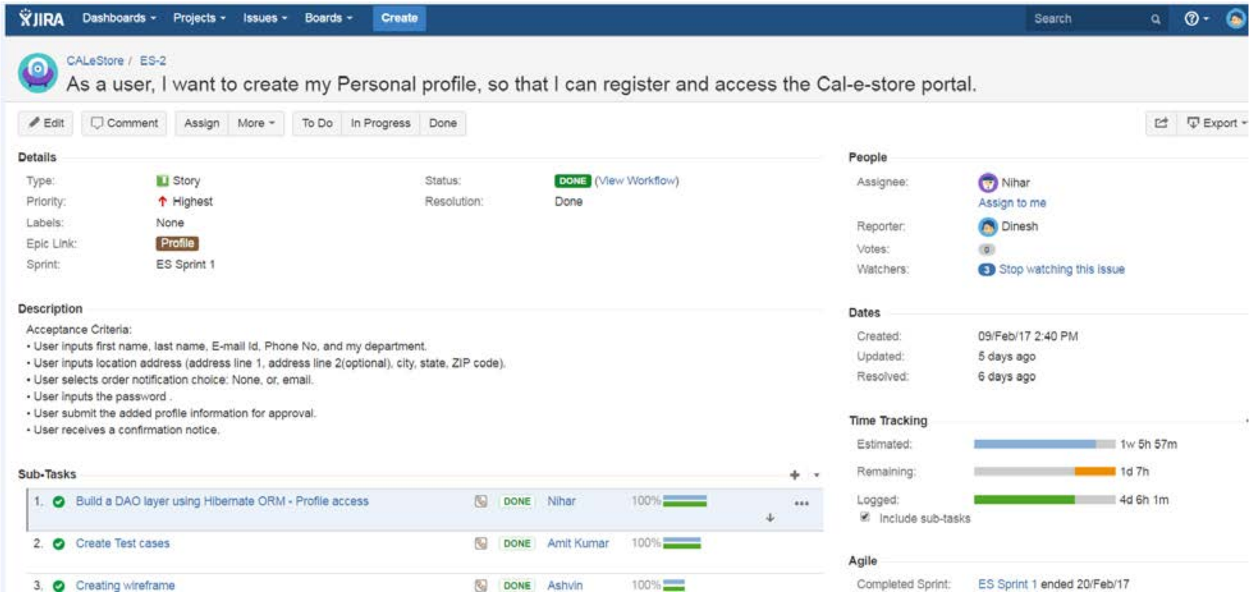
Check out

Framework
ES-99 Application Framework
Issues 3
Completed 0
Unestimated 0

ES Sprint 1 11 issues 39 13 0

13/Feb/17 9:00 AM - 17/Feb/17 11:59 PM

Key	Summary	Epic	Estimate
ES-2	As a user, I want to create my Personal profile, so that I can register and access the Cal-e-store portal.	Profile	5
ES-3	As an authorized user, I want to login, so that I can access the User Account	Profile	3
ES-4	As an authorized admin, I want to login, so that I can access Admin account.	Profile	3
ES-5	As an authorized admin, I want to manage user registration request, so that I can control the user accounts.	Profile	3
ES-9	As an authorized user, I want to view the product Catalog, so that I can view all the published Products and Services in the portal.	Catalog	5
ES-10	As an authorized user, I want to view the product details, so that I can view all the specifications to make a buying decision	Catalog	6
ES-19	As an authorized user, I want to compare the products, so that I can make decision on the quality and price of the Products and Services.	Catalog	5
ES-22	As an authorized user, I want to check out, so that I can order and get my products shipped to my address.	Check out	5
ES-100	As a System, I require a modularised Application Framework, so that developers can work independentaly	Framework	8
ES-101	As a System, I want to set up a CI, CD and CO pipeline, so that the software quality is ensured	Framework	5
ES-102	As a System, I require product data to be migrated to database, so that application can consume the product database	Framework	5



CALeStore / ES-2

As a user, I want to create my Personal profile, so that I can register and access the Cal-e-store portal.

Details

Type: Story
Priority: Highest
Labels: None
Epic Link: Profile
Sprint: ES Sprint 1

Status: DONE (View Workflow)
Resolution: Done

Description

Acceptance Criteria:

- User inputs first name, last name, E-mail Id, Phone No, and my department.
- User inputs location address (address line 1, address line 2(optional), city, state, ZIP code).
- User selects order notification choice: None, or, email.
- User inputs the password .
- User submit the added profile information for approval.
- User receives a confirmation notice.

Sub-Tasks

1.	Build a DAO layer using Hibernate ORM - Profile access	DONE	Nihar	100%
2.	Create Test cases	DONE	Amit Kumar	100%
3.	Creating wireframe	DONE	Ashvin	100%

People

Assignee: Nihar
Assign to me
Reporter: Dinesh
Votes: 0
Watchers: 3 Stop watching this issue

Dates

Created: 09/Feb/17 2:40 PM
Updated: 5 days ago
Resolved: 6 days ago

Time Tracking

Estimated: 1w 5h 57m
Remaining: 1d 7h
Logged: 4d 6h 1m
☑ Include sub-tasks

Agile

Completed Sprint: ES Sprint 1 ended 20/Feb/17

Daily Stand-up

During sprint execution, the team had daily stand-up meetings for a maximum duration of 15 minutes. These sessions helped the team remove impediments to development and promoted quick decision-making to improve understanding of the project status.

Sprint Review

The review for the first sprint was held at the end of the sprint to inspect the Increment and adapt the Product Backlog. All project stakeholders were invited by the Product Manager. The Product Manager outlined the Definition of Done and which Product Backlog items had met the criteria of being Done. The development team demonstrated the work that was Done in the Increment. The Product Manager then discussed the product backlog status and projected a likely completion date. A user story that had some feedback from the Product Manager was marked as “Not Done” and returned to the backlog. The group collaborated to identify next steps for subsequent sprint planning.

At the end of the review, the Product Backlog was revised to include probable Product Backlog items for the next sprint.

Sprint Retrospective

The team met to discuss the finished sprint and create a plan for improving the next sprint. Members identified what worked well and where improvements could be made. The team prioritized improvement areas and decided to implement ideas for improving the top three items during Sprint 2.

Sprint 2

Based on review comments from Sprint 1, the Product Manager reprioritized the Product Backlog to prepare for Sprint 2. The theme for Sprint 2 was “Added Functionality” to the MVP product created in Sprint 1.

The team began Sprint 2 with a sprint planning meeting. The entire process was repeated similar to Sprint 1.

The planning poker method was used to estimate the sprint candidates. The team committed to eight user stories that aligned with the theme of the sprint. Respective tasks with estimates were created for each user story. The sprint was then started in Jira and the team began working on the tasks.

Daily stand-up meetings continued to occur as in previous sprints.

Sprint 2 concluded with the Demo and the Sprint Retrospective. The team was able to complete the Sprint 2 commitment and all the user stories were accepted by the Product Manager. Feedback on improving the user experience was noted for implementation in Sprint 3. The Product Manager then listed out the remaining user stories to be implemented in the final Sprint.

The Sprint 2 retrospective gave the team a chance to reflect on how Sprint 2 fared and to discuss the action items for the final Sprint.

Sprint 3

Sprint 3 started with the Sprint Planning meeting. The Product Manager explained the remaining user stories in the Backlog. The theme for the final sprint was “Reports and Dashboard”. Five stories pertaining to the reporting functionality were pulled into the Sprint 3. The team estimated their commitment and added the respective tasks for the user stories. JIRA was updated to start the sprint and the team commenced working on their respective user stories.

Daily stand-up meetings continued to occur as in previous sprints.

The final sprint was demonstrated to the stakeholders that included the end-to-end functionality developed for the prototype. The Product Manager explained that the mandatory user stories for the prototype was complete and some ‘nice to have’ features were left in the backlog in case there was scope to improve the product at a later stage.

The sprint concluded with the Retrospective and pizza party to celebrate the successful completion of the prototype.