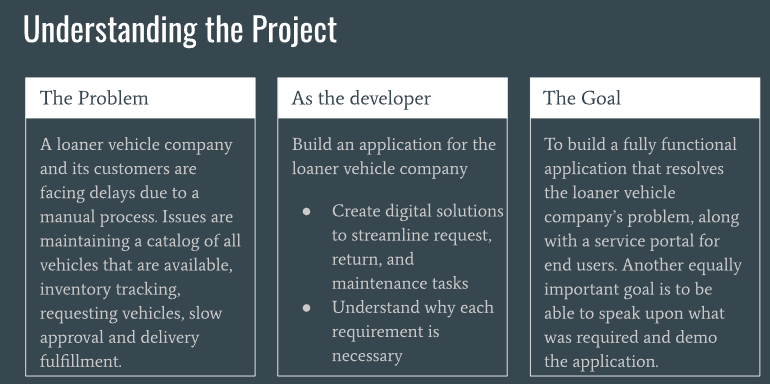
**Project**

**Lesson – 1**

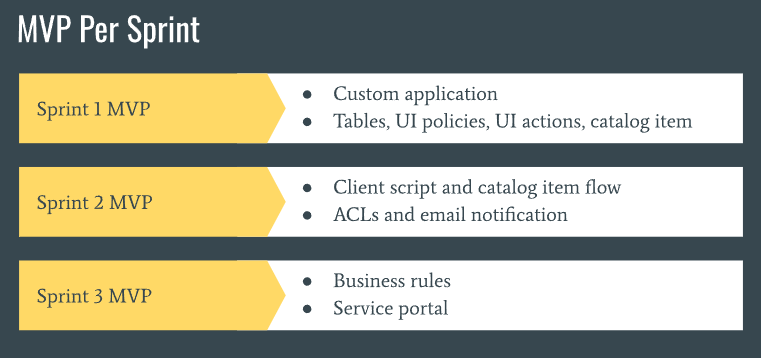
**Project overview video + slide deck**

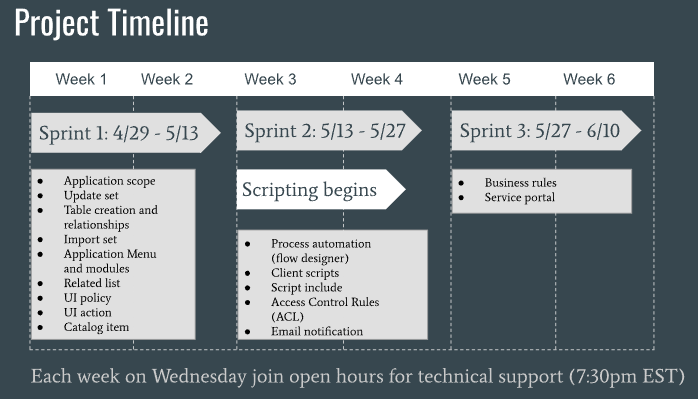


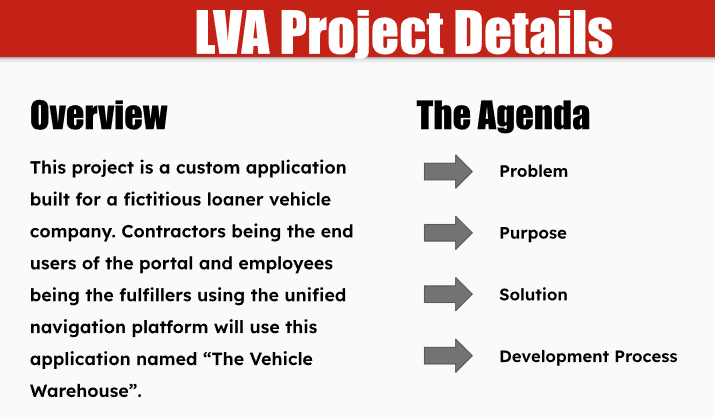


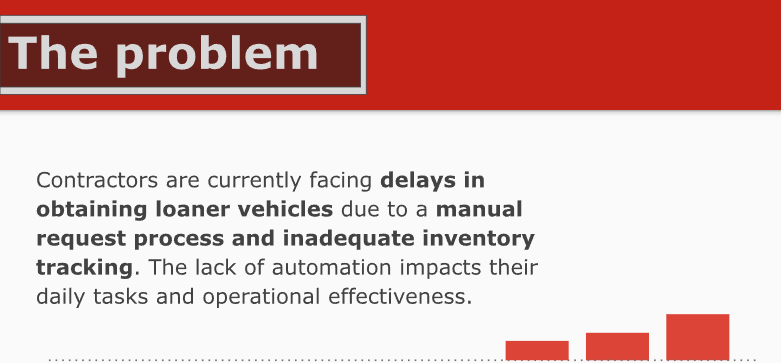


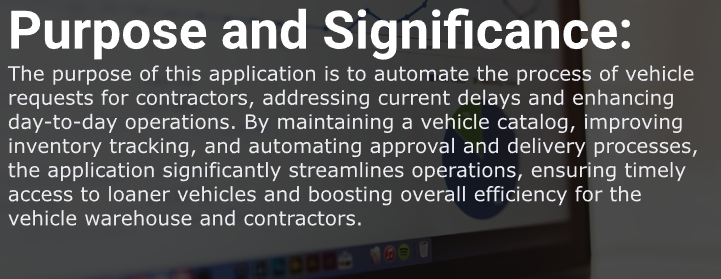




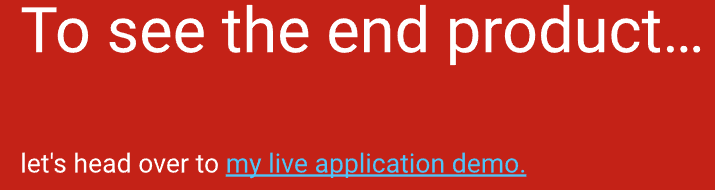


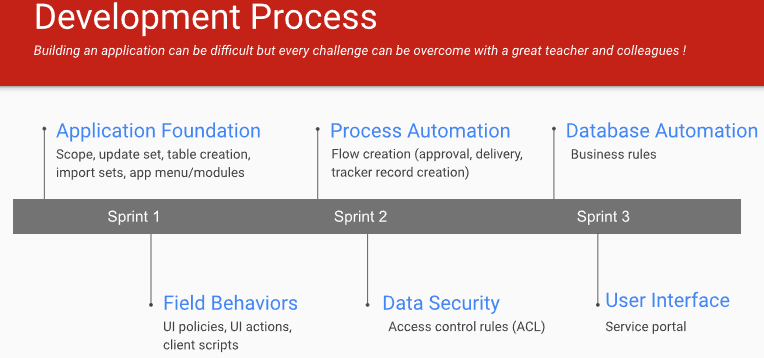














Lesson -2

**Project overview document:**

The business requirements and goals involve creating an automated process for requesting loaner vehicles through a self-service portal to replace the current manual process. This change is sought to address delays in obtaining vehicles, which hinder contractors’ daily operations. The proposed project will manage vehicle inventory, track availability, and streamline the request, approval, and delivery process into a single flow.

**Loaner Vehicle Request and Inventory Management Project**

**Business Requirements and Goals**

The various contractors who need a loaner vehicle would like to be able to automate the process of requesting vehicles from the vehicle warehouse. They would like to move away from the current manual process as it has been a struggle to get what they need in a timely manner which directly affects their ability to perform their day-to-day activities.

**Functional Overview**

The proposed Loaner Vehicle Request and Inventory Management Project aims to solution current issues of maintaining a catalog of all vehicles that are available, inventory tracking, and will also introduce a process that incorporates approval and delivery tasks into one flow.

**Technical and Process Overview**

**Phase 1:** Application Foundation - Field/Form Behavior - **Request Process**

The request process is set off by users (contractors) **requesting for a vehicle of their choice**. An intake form that has all the necessary fields for the request will be available on the Service Portal. The intake form will be asking for the following information.

1. Vehicle Selection from a list of available vehicle records stored on the system.

2. The date they would need to receive the vehicle.

3. The date they expect to return the vehicle.

4. Office Location.

5. Delivery information.

6. Additional comments.

Once the request is submitted, an automated flow will trigger an approval to an approval group Upon approval, the state field of the selected vehicle record will be set to unavailable and a new entry on the Vehicle Tracker table will be created. The Vehicle Tracker entry will be populated with the following information: vehicle borrowed, the person who made the request, location, start and end date and other important information. A task to deliver the vehicle to the contractor will be created. Once the delivery task is closed the request ticket is closed.

-we will create **\*Service Catalog, Catalog item, Create a Flow**

**Things to be accomplished in sprint 1**

- Application scope

- Update set

- Table creation and relationships

- Import set

- Application Menu and modules

- Related list

**Phase 2:** **Return Process** - Process Automation - Data Security

Once the vehicle is returned, the **warehouse team is responsible** for **assessing the vehicle** and filling in a section on the Vehicle Tracker record that mostly asks information on the **condition of the vehicle**.

If the vehicle is in good condition, the warehouse team will need to click the Return to Warehouse button which will set the Vehicle record to Available. If the vehicle needs to be serviced, then they would need to click the Send to Repair button which will set the Vehicle record to Maintenance and will kick off the Maintenance process.

We will create:

- UI policy

- UI action

**Things to be accomplished in sprint 2**

- Process automation (flow designer)

- Client script

- Script include

- Access Control Rules (ACL)

- Email notification

**Phase 3:** **Maintenance process** - Database Automation - User Interface

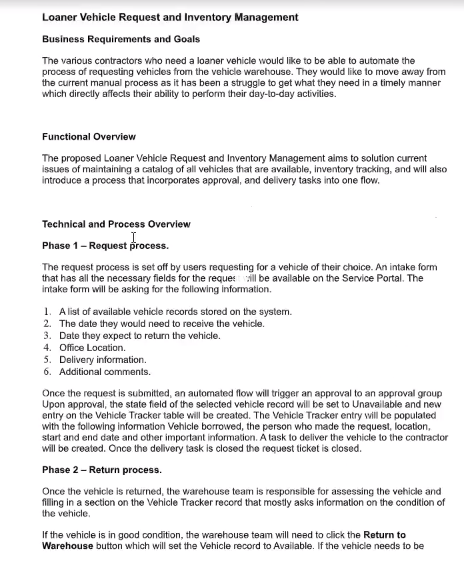
The maintenance process triggers a Vehicle Servicing record that will serve as an official document to work on the vehicle. The maintenance task will require the team to document what the issue is on the vehicle and what they did to resolve it. Once they close the task, they determine if the vehicle can be placed back to the warehouse or if it will need to be decommissioned.

**Things to be accomplished in sprint 3**

We will create:

- Reference Fields, Business rules

- Service porta

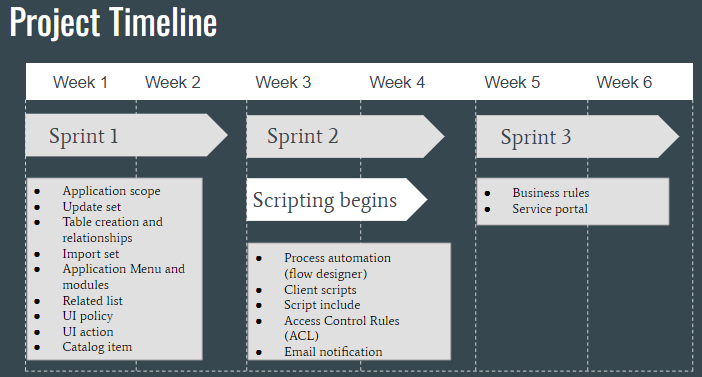


Lesson – 3

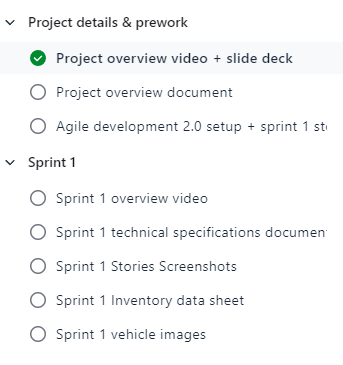
**Agile development 2.0 setup \_ Sprint 1 stories**

**Hands-On App Dev Projects**

1. You should have either completed the course pre-work or be working on installing Agile Development 2.0, setting up your Agile board, and importing sprint 1 stories(which is what the pre-work entails). Starting today, you will begin working on the stories in sprint 1(as seen in the attached screenshot).



2.



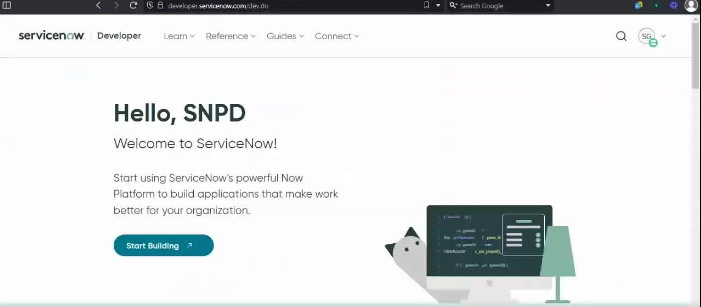
3. **ServiceNow Product Development**

1. Download the Agile Development 2.0 plugin and Agile Development - Unified Backlog
2. Check out the stories table and the data your importing - where you’re loading the data
3. Create a product record for the loaner vehicle application
4. Import sprint 1 stories
5. Create an agile group and add your admin user to it
6. Assign all of the imported stories to the agile group created
7. Go to the agile board and set it up
8. Create sprint 1
9. Start the sprint

When sprint 2 starts import that file into the story table and create a sprint for it in your agile board

Lesson - 3

**Downloading Agile development 2.0 downloading plugins:**



In app engine studio

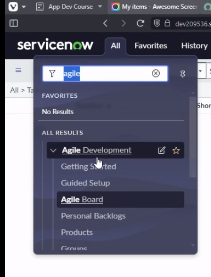
**To see Agile Development Modules and Applications Download the Following Plugins**

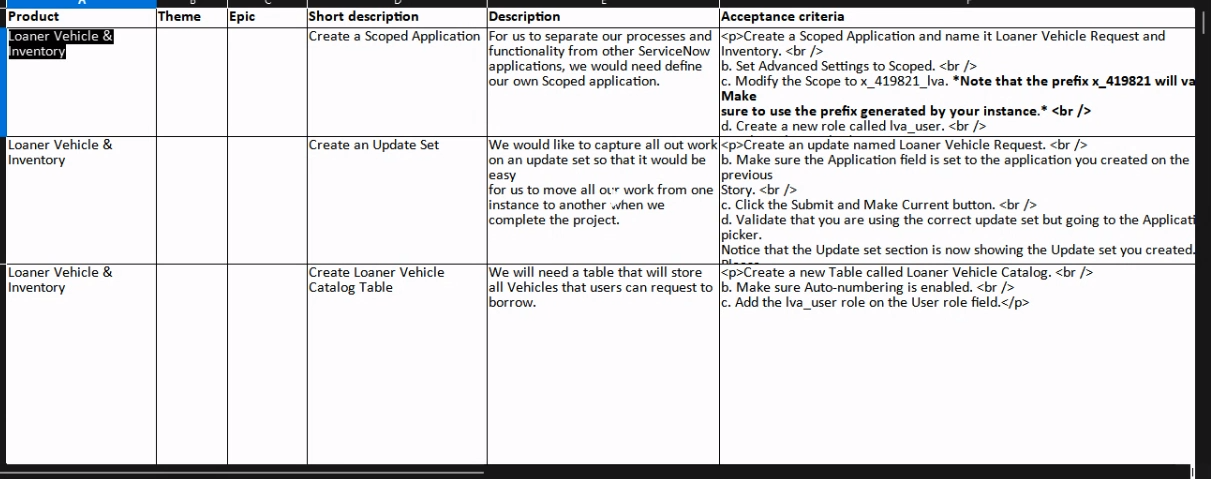
**Step - 1**. **Plugins** - A**ll >plugins>plugins>in search type in agile dev>** scroll down to **ServiceNow Products > scroll down and download below 2 plugins**

**1. Agile development 2.0**

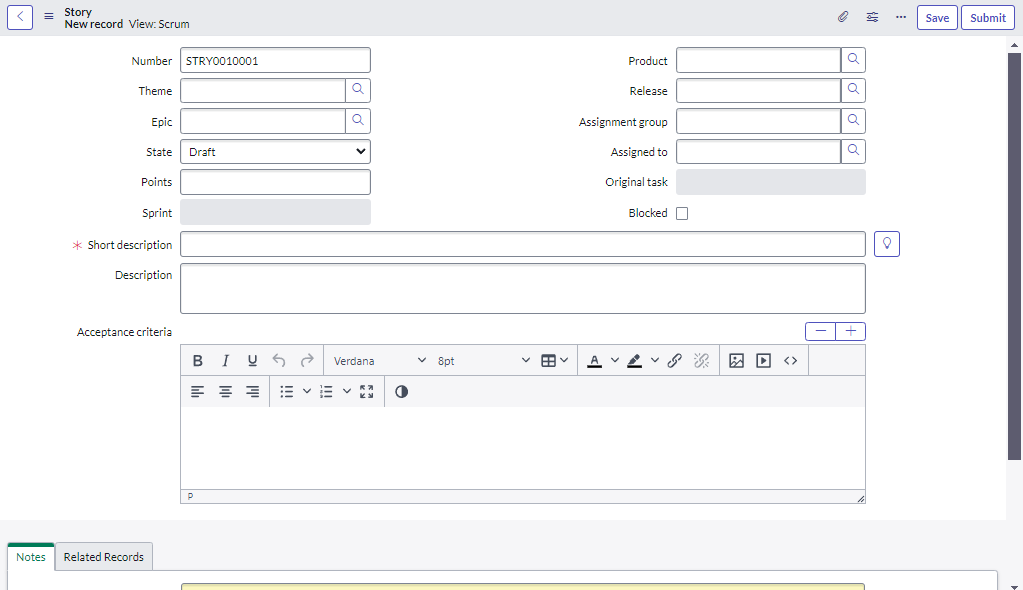
**2. Agile development Unified backlog once downloaded**

**Step - 2. Stories** - **All > agile >agile development> Stories(table)> New**





Story Form:



**Create Product table:** (**Loner Vehicle & Inventory** Record)

All> Agile development> Product > Enter Product Name as: **Loner Vehicle & Inventory > Submit**

**Import Story:** to input go back to Story Table: All> Stories > right click on three dots of any column > Import> uncheck the (create and excel file to input data) > choose file (Sprint 1 stories(1).xlsx) > Upload > Preview Import Data > Complete import

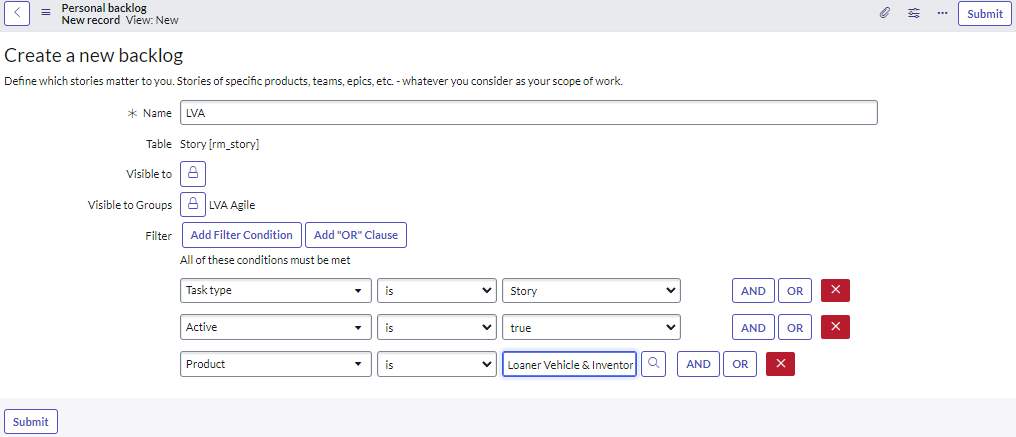
You will see 14 stories

**Create Agile Group: step 1.** To access the agile storiesAll> Create Agile Group > Name: LVA Agile > Submit

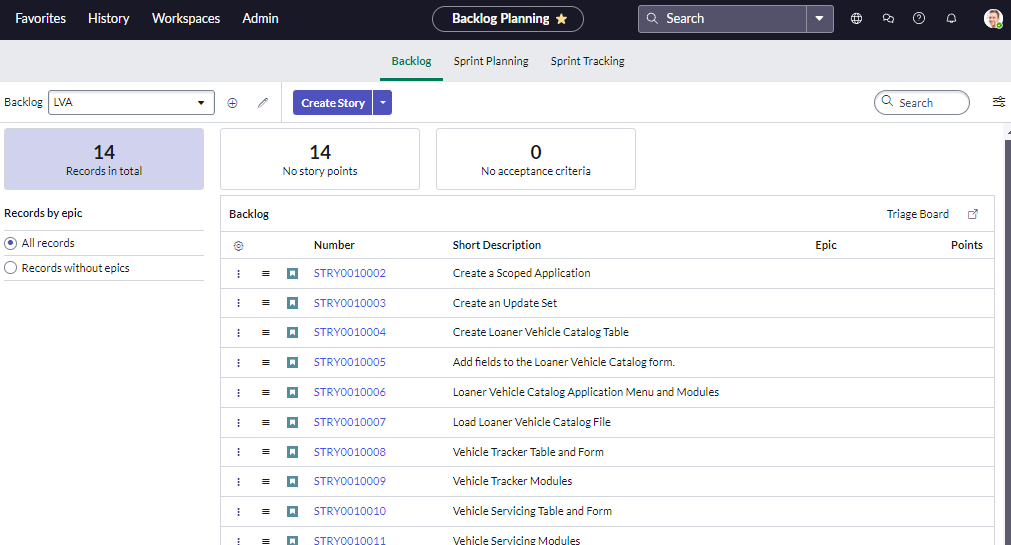
**Add A User to Agile Group step 2.:** scroll down while adding **User** to the group as Sys Admin >click Edit> type in System Admin> Double click > Save > Right click on menu and SAVE.

**Agile Board:** All> Agile Development > Agile Board > Create Backlog > Name: LVA > Visible to groups: LVA Agile >click on Lock symbol > Submit

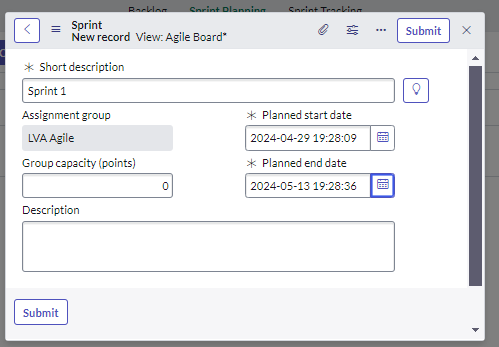
Agile Board:



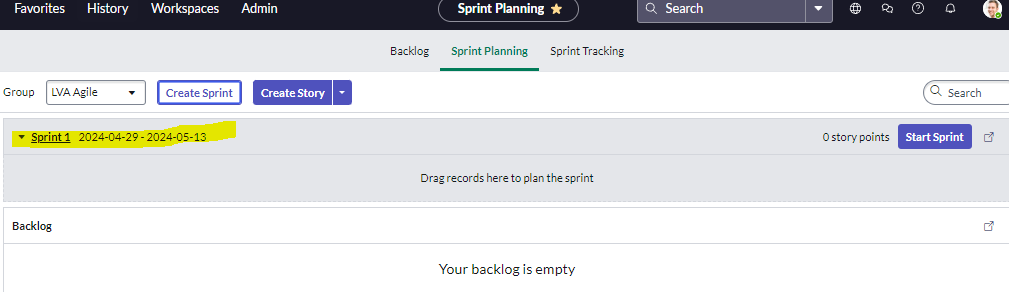
Once submit we can see below 14 stories in our backlog:



**Specifying the 14 stories belong to Sprint -1:** Click on **Sprint Planning Tab > Click Create Sprint >** Enter short Description - Sprint 1 > Planned start date 29/04 and tick > Planned end date 13/05 and tick > Submit

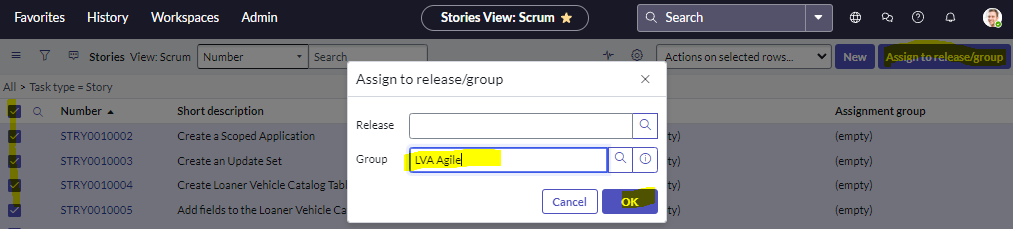


Once Submit you can see sprint 1



Now **Assign LVA agile group** to all the stories that we imported:

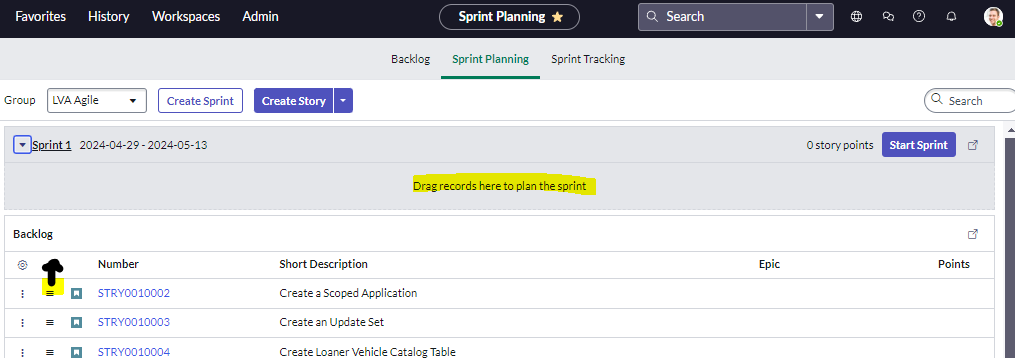
Goto All> Stories> select all the Stories – Tick all checkboxes> Assign to release / group>



So now all stories been added to **product** and **assignment groups**.

**Agile Board:** To add Story to Sprint -1

Go back to All > Agile Board > Drag and drop these stories to sprint-1 in Ascending order.



\*Hover over on (Hamburger Icon) and drag.

**Import Sprint-2 Stories:**

All> Stories > right click on three dots of any column > Import> uncheck the (create and excel file to input data) > choose file (Sprint 2 stories.xlsx) > Upload > Preview Import Data > Complete import

Sprint-2 has only 4 stories

Stories assigned to your backlog you can see them here:

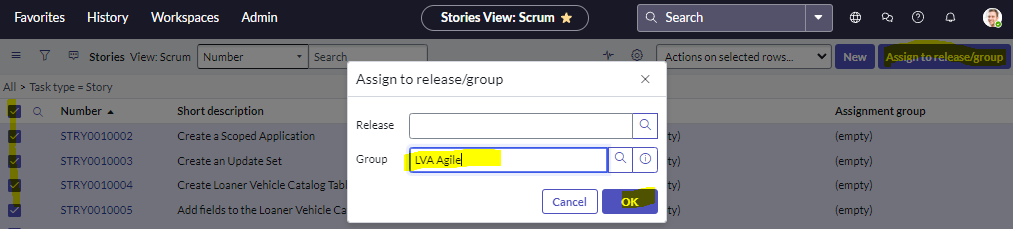
Then go to All > Agile board > Backlog >scroll down ad see all 4 stories been added to Backlog.

We can’t see them in sprint planning until we add them to Agile Group.

**Assigning Stories to Assignment Group using** Filter condition Assignment group is empty: Click on All> Stories> Click on Filter icon > **Assignment Group is Empty >** Run

Now **Assign to LVA agile group** to all the stories that we imported:

Goto All> Stories> select all the Stories – Tick all checkboxes> Assign to release / group>



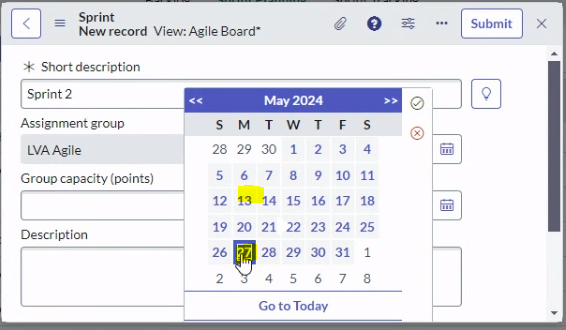
So now all stories been added to **product** and **assignment groups**.

Stories assigned to your backlog you can see them here: for Sprint-2

Then go to All > Agile board > Sprint Planning >scroll down and see all 4 stories been added your backlog.

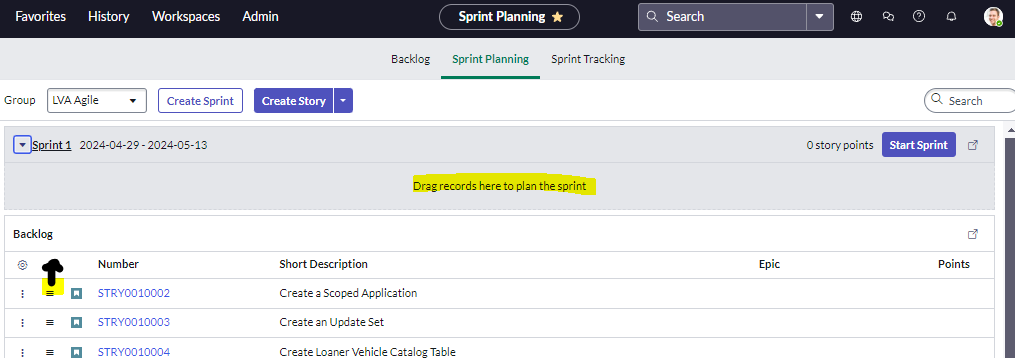
To Make sure to create another Sprint:

Click on Create sprint Tab > Short description- Sprint 2 > then choose the start date as 13 and end date as 27 > click on green tick > submit



**Agile Board:** To add Story to Sprint -2

Go back to All > Agile Board > Drag and drop these stories to sprint-1 in Ascending order.



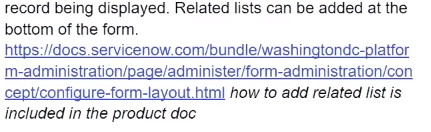
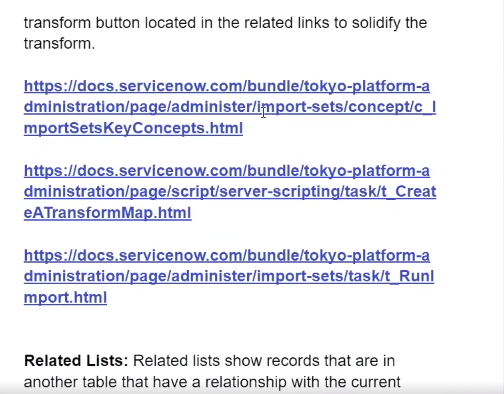
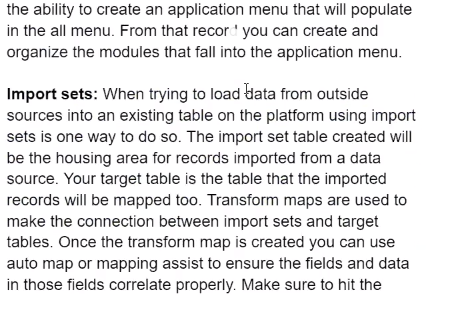
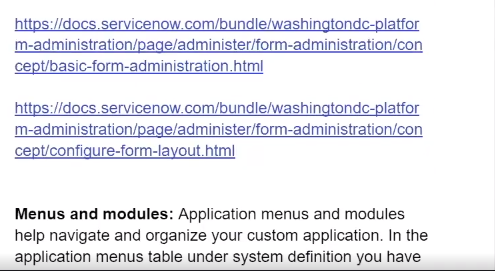
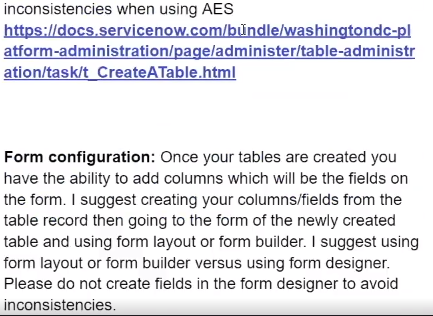
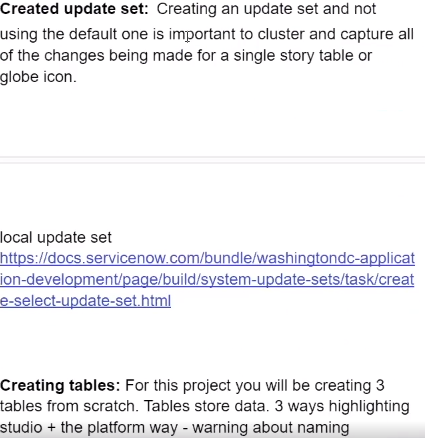
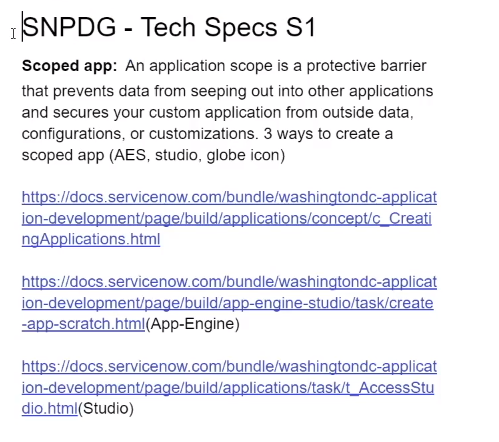
\*Hover over on (Hamburger Icon) and drag.

-END-

Lesson 4

**Sprint 1 Part-1 and Part-2**

**Sprint Over View**



Sprint -1 Lab

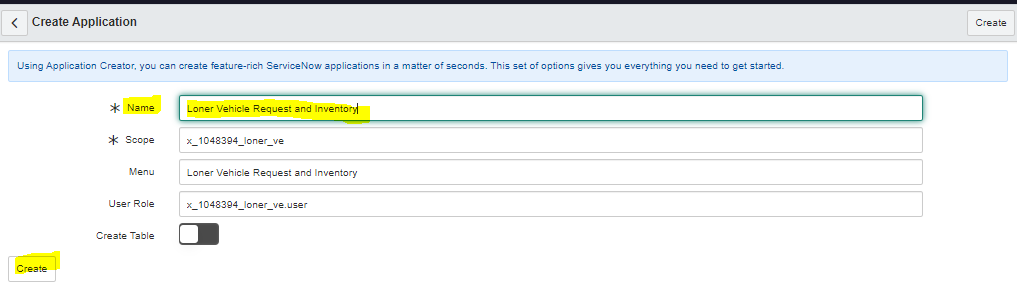
* 14 stories
* Application Scope, Update Set
* Create Tables
* Configure Forms
* Create Menu and Modules for each

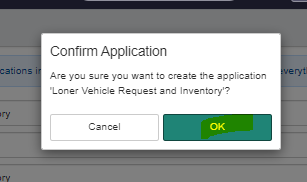
1. **Application Scope:** is the protect the barrier, data out other applications and secure our own custom applications from outside data, configurations and customizations.

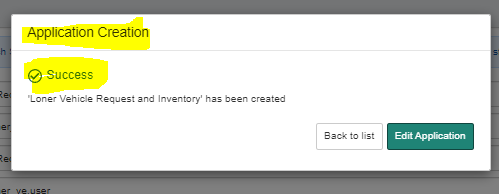
**3 ways to do this:**

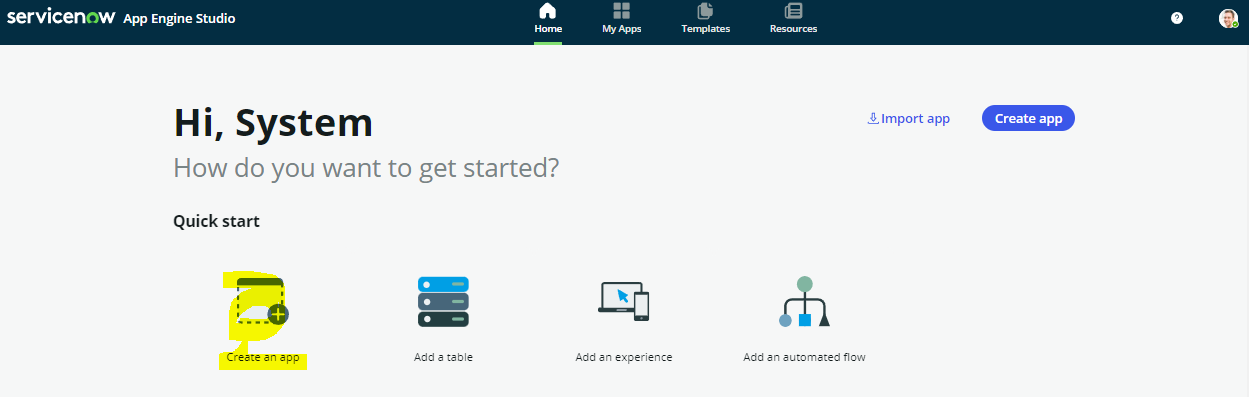
1. Click on the **Globe Icon > application scope** click on **the arrow (>) >** click on **open List icon** **> New >** choose custom applicationsclick on **Create**
2. **From the App Engine Studio >** Click on **Let’s Go >** Click on **Create an App**
3. **All > System Applications-Studio >** Click **Create Application**

1. **Globe Icon > application scope** click on **the arrow (>) >** click on **open List icon** **> New >** choose custom applicationsclick on **Create**

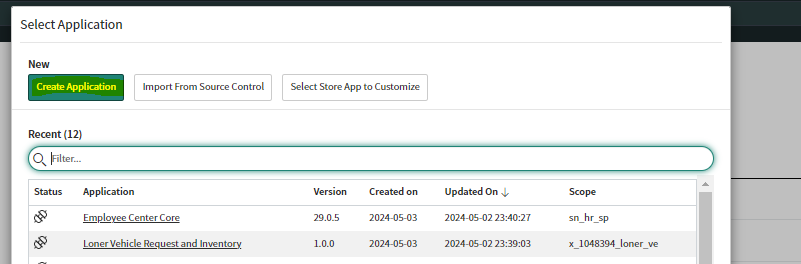






2. All > App engine Studio Lets Go

3.All > system application > studio> Create application



1. **Update Set**: You can create update set for each story or whole application in one update set.

**2 ways**:

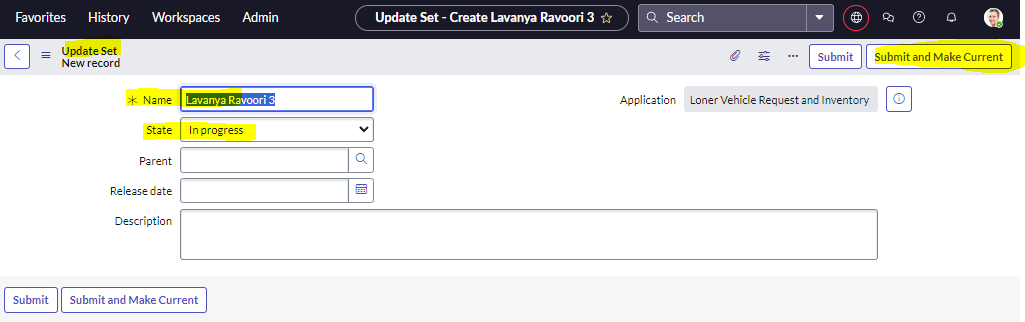
Click on the **Globe icon** > Update set> click on

1. List icon or+ icon >

2. **All** > **Local update set** under system update set

> Click on **New icon >** set state **In progress** >click **Submit make current**

\*Always make use while creating update set state always must be In progress



Always you log in make sure **Application Scope** - Loner Vehicle Request and Inventory

**Update Set** – Lavanya Ravoori (Loner Vehicle Request and Inventory)

**3. Tables:**

**A table is a collection of records in the database**. Each record corresponds to a row in a table, and each field on a record corresponds to a column on that table. **Applications use tables and records to manage data and process.**

Tables are to store data for you for this application we are going to create

3 tables from Scratch, ways to create table.

1. All > System Definition >Tables > New > Create
2. App Engine Studio > Lets go> Add a table

**4.Form:**

Menu:

