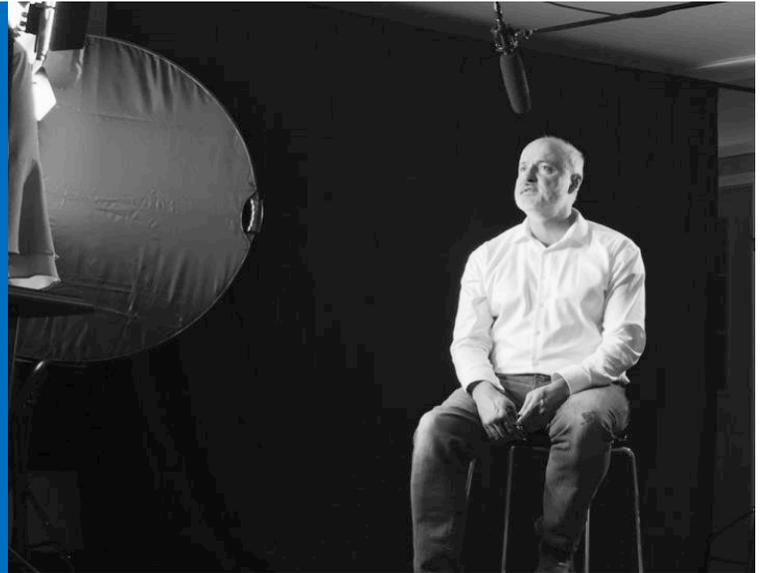# Launch Your Salesforce Career Bootcamp

Module 5:
Agile Enhancements App Pack

Table of Contents

Lesson 1 – Agile Application Enhancements Pack Introduction

In this introductory lecture, I discuss what lies ahead for this module – the Agile Applications Enhancements Pack.

In this module, we will revisit the apps we previously built. We identify areas for improvement and then work through making those improvements.

Let's get started with improving our applications!

Notes:

_____

_____

_____

_____

_____

_____

_____

Lesson 2 – Entering Job Opportunities into Our App

In this lesson, we use the app and test it out – much like real world user acceptance testing to find limitations and note enhancements.

Remember from our Job Opportunities App, that we have structured all job opportunities to start out as a Lead.

How do job leads come in?

- Inbound inquiries
    - Calls
    - Emails
    - Texts
    - Tweets
    - Direct Messages
    - Networking / Meet someone
- Outbound inquiries you initiate
    - Applying on a company website
    - Applying on a job board
    - Email
    - Call
    - Networking / meet someone

In this lesson, we revisit our Job Opportunity App, and incorporate populating jobs.

Now try doing the same from the Mobile app – is your app in the mobile app?

Notes:

_____

_____

_____

_____

_____

_____

Lesson 3 – Creating a Lightning App Page for Mobile

In this lesson, we create a Lightning App Page for the Jobs & Recruiters application. We make the app page available in Lightning Experience and Mobile.

Notes:

_____

_____

_____

_____

_____

_____

Lesson 4 – Creating Lead Record Types

In this lesson, we introduce Record Types to the Lead object. These record types include:
- Job Lead
- Project Lead
- Volunteer Lead

We also create a Lead Process for Volunteer Leads

We adjust the Lead Source selections, based on Record Type. We add custom fields to capture projected Salary information on job leads.

Notes:

_____

_____

_____

_____

_____

_____

_____

Lesson 5 – Mileage Tracker Enhancements

In this lesson, we work through improvements to Mileage Tracker App. We explore the use case of not only entering mileage for trips for ourselves, but how to solution trip entry for other people. We then create a Lookup field from Trip to Contact.

To ensure data quality, we then create a Lookup Filter on the Contact field to make sure we only select Contacts that belong to the Account that the trip is being logged for.

If we are entering a trip for someone other than ourselves, we are going to elect to introduce record types on Trip for either self or other and then make default record type self and change the page layouts on trips so that the Contact Lookup field is not on the Self trips.

We discuss how to edit Page Layouts. We also look at setting Record Types.

Resource:
Master Record Type Explained - https://youtu.be/GKu2J2Ij1NE

Notes:

_____

_____

_____

_____

_____

_____

Lesson 6 – Logging in as Another User

In this lesson, we test out our Mileage Tracker enhancements by logging in as another user. To enable this, we access **Login Access Policies**. We then log in as another user and log a trip to verify that they only have the ability to enter in a Trip with a Record Type of **Self** and not **Other**.

We also disable the **Force Re-Login After Login-As User** setting in *Session Settings*. This keeps us logged in as an admin whenever we log in and then out as another user.

While logged in as another user, we note a few fields on the Trip page layout that were missing. This is due to the President Profile not having access to the Project object. We then adjust the profile object settings to verify that the Project lookup now displays in the Trip page layout for the other user by logging in as them.

Resource:
Enabling Admins to Login as Any User -
https://help.salesforce.com/articleView?id=Enabling-Organization-Admins-Can-Login-as-Any-User&type=1

Notes:

_____

_____

_____

_____

_____

_____

Lesson 7 – Expense App Enhancement – Profile Specific Validation Rule

In this lesson, we create a profile-specific validation rule. This is to enhance our Expense Tracker application. We work through the scenario of having only administrators able to enter an expense that is greater than $25.

Validation Rule Code on Expense Object:
AND(Amount__c > 25,$Profile.Name <> 'System Administrator')

Notes:

_____

_____

_____

_____

_____

_____

_____

Lesson 8 – Time Tracker Enhancements – User Specific Validation Rule

In this lesson, we improve the Time Tracker application by first adding a Status picklist field to allow for submitting, approving and rejecting Time Cards. We then proceed with enabling Field History Tracking and setting the fields to track. We also adjust the Chatter Feed Tracking to now track the Status picklist changes as well in the Chatter Feed of Time Cards.

We then create a more complex validation rule that is user specific. In our scenario, we wish to block the owner of a Time Card from Approving or Rejecting their own Time Card. This process calls on us to also adjust the President Profile Object Settings to grant further access rights to both the Time Entry and Time Card object.

You could further extend the functionality of this application with additional validation rules, such as not allowing the owner of a time card from changing the Total Hours of a Time Card once the Status has been set to Approved, for example.

Validation Rule Code on Time Card Object:
AND (OR (ISPICKVAL (Status__c, 'Approved'), ISPICKVAL(Status__c, 'Rejected')), Owner:User.Id = $User.Id)
Notes:

_____

_____

_____

_____

_____

_____

Lesson 9 – Security Enhancements – Profiles and Permission Sets

In this lesson, we take away the ability of those with the Profile of Vice President to delete accounts. We then make an exception for one of those Vice President's by creating a Permission Set and assign it to that user.
.

Notes:

_____

_____

_____

_____

_____

_____

_____

Lesson 10 –Module 4 Assignment Explained

I wanted to be sure you didn't miss these important videos that detail how to complete the Module 4 Assignment and Stretch Assignment. In this first video, I run through how to complete the previous module's assignment.

Notes:

_____

_____

_____

_____

_____

_____

Lesson 11 –Module 4 Stretch Assignment Explained

In this lesson, I cover the majority of the previous module's stretch assignment, except for the Lead Conversion step, which I cover in the next lesson.

We create a validation rule to require that the Jobs Website field be populated whenever the Lead Source field equals web. The error code formula for this validation rule is:

AND(ISPICKVAL(LeadSource,'Web'), Jobs_Website__c = '')

Notes:

_____

_____

_____

_____

_____

_____

Lesson 12 – Module 4 Stretch Assignment Lead Conversion

This is the final lesson related to the run-through and explanation of the previous module's assignments. This lesson covers the final step of last module's stretch assignment related to the Lead Conversion process.

Notes:

_____

_____

_____

_____

_____

_____

Module Review and Assignments

We covered a lot of ground in this module – the Agile Applications Enhancements Pack. We created a Lightning App Page and made our Jobs & Recruiters app mobile. We introduced Record Types to the Lead Object.

We also enhanced our Mileage Tracker application, relating Trips to Projects via a Lookup field. We also enabled entering mileage for other people via a Lookup field on the Contact object, and incorporated Lookup filters. We introduced Record Types on the Trip object as well.

We also created several Validation Rules in this module, such as limiting the functionality of entering Trips for others to only be available to Admins. Other validation rules were created to limit expense entries to a certain amount for non-admins, as well as preventing Time Card owners from changing the Total Hours of a Time Card once its Status is Approved.

We also set the ability to log in as other users and tested usability and visibility for other users across our various apps. This lead to key discoveries related to missing fields, setting Field Level Security, adjusting page layouts and more.

We also blocked the ability for certain users to be able to delete account records. We then make exceptions to grant delete rights to certain users via a Permission set.

We also looked at the previous module's assignments and ran through how to accomplish those tasks. These tasks included creating another custom object called Jobs Website. We also created custom URL fields to capture each Job Website homepage as well as recording the URL of your Profile on each of the Jobs websites that you have posted your resume to.

We then leverage this new Jobs Website object on the Lead process. We create a custom Lookup field on the Lead object to Lookup to the Jobs Website object. We then create a corresponding Jobs Website field on the Opportunity object and then set up the lead mapping so that the Jobs Website carries over from Lead to Opportunity during the Lead Conversion process.

We also create a validation rule so that if the Lead Source is Web then the Jobs Website field is required on the Lead record. This requires using the ISPICKVAL function in the validation rule along with an AND function.

Now it's time to apply the skills that you've acquired during this module by completing the assignment(s) below.

Module Assignment

Now that we have a Jobs Website custom object and we leverage those records when entering new Leads that come in from a Lead Source of Web, you can remove the Lead Source picklist values for Job Leads.

Do not delete those picklist values, because that would lead to data loss on existing records. Also, you may wish to specify a Lead Source of LinkedIn or Monster, or Indeed, or Career Builder, for example on Leads other than Job Leads, i.e. Project Leads or Volunteer Leads. How would you go about removing Lead Source picklist values for one Lead Record Type but not others?

Stretch Assignment

In using the Jobs & Recruiters application further, and entering in Job Leads, and converting them into Job Opportunities, along with Accounts and Contacts as part of the Lead Conversion process, you may desire greater functionality with your app in its current state.

Once you have converted a Job Lead into a Job Opportunity, what is the usability like when entering in a Resume Submission? It may be helpful to reflect the Lead Source on the Resume Submission object.

How can you pull in the Lead Source information on the Resume Submission object?

Whatever your proposed solution is, build it out in your own org.

We will be discussing this further on this week's group coaching call.

Notes and Action Plan