United_Solar_Project_Requirements

Project 2

Directions: Each associate must choose a variety of use cases from the available categories. Submit to your trainer this document filled out with the associate assigned to each use case. Each use case is assigned a number of points and each category has a maximum number of points that can come from it. In order for each team member to receive 100%, the team must score 100 points collectively. Any individual variation in grades is based on presentation ability. For larger use cases, multiple associates can be put as having responsibility. Individuals may not select use cases from only a single category and should attempt at least one from each category if possible. These are aligned with the certification categories and a well rounded understanding of them will aid you in your certification efforts.

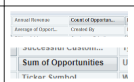
Analysis: Before starting work on the project, the team must perform analysis on the use cases. First, an ERD must be crafted to support the needs of FBC mentioned below. In addition to this ERD, the team must submit a document analyzing the chosen use cases and the best tool to complete the use case. Compare the available tools and select the most appropriate one.

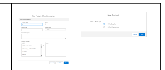
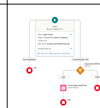
Business Overview: FBC is a major office supply company. They mostly work with and supply government agencies. FBC not only supplies things like copy paper, but they also offer infrastructure as well - installing servers and desktops for these government organizations. FBC is moving towards Salesforce to help them manage the business. Your team will be responsible for customizing their Salesforce instance and meeting their business requirments.

FBC's Accounts, as previously mentioned, are usually government ogranizations. They need to keep a list of Contacts at these organizations to close any potential deals. New Leads should also be tracked. The list of Contacts should also include service technicians who work for FBC. When a service technician needs to go install hardware for an Account, a Work Order should be logged. Further, all Order information should be stored in the database. This includes which Products were in the Order. When our Accounts give feedback and tell us we did a great job, we should track that feeback in the form of customer success stories. Customer success stories should include a description of the positive feedback and a reference to the product that helped our customer achieve their goals.

Beyond implementing the above requirments, FBC would like your team to create an Employee Hub written in Visualforce. This should be a one stop shop for managing all of the Visualforce related feature requests below.

FBC

Category	Use Case	Associate	Points	Progress	Comments	Relevant Names	Images
Developer Fundamentals Max Points: 23							
	Implement the ERD. Add in custom fields you think would be useful to track on each object.	**MANDATORY, ALL ASSOCIATES MUST ASSIST IN THIS USE CASE**	5		Completed and implemented as it stands, but as we continue working, we may need to add more.		
Validation Rule	When capturing phone numbers into the database, we need to ensure that the data follows a USA phone number format.	Jason	3		Completed, Used validation rules and added to all phone number fields for contact. Used regex in the formula field for specific format.	5 validation rules on contact -phone -mobile phone -other phone -asst phone -home phone	
Vlookup formula/Maybe seperate custom object	FBC's service technicians service a variety of zip codes, but occasionally when technicians are too occupied certain zip codes go out of service temporarily. The zip codes and whether or not they are in-service needs to be maintained by the technician managers, who are not admins. **Ensure that users don't save work orders with zip codes that aren't currently in service.**	Evgeny	6		Complete, Technician Manager permission set available for assignment. An In-Service Zip Codes custom object was made along with a vlookup validation rule on the Work Order standard object.		
Flow or Apex if conversion requires creation of the 3 object records	FBC would like to convert all Leads within a month of being added to the database. All displayed Leads should show a date that represents this suggested conversion date.	Anyone	3				
Formula + Roll-up summaries	Sales team members would like to display the average amount of all related Opportunities on each Account record.	Richard	3		Complete.	Custom fields on Accounts: "Count of Opportunities", "Sum of Opportunities", "Average of Opportunities".	
Custom object Related List	Whenever an Account is displayed, we should show the number of customer success stories associated with the Account.	Arashdeep	3		Complete. Created MD relationships with customer success stories and Account(parent). Created a roll-up summary on Account with filter critera to get only the sucessfull stories	Custom Object: Customer Feedback MD with Account CustomFields: Sucess and Description Account Field: Successful Customer Story	

Record Types (Products vs Services)	We should differentiate between our two product categories - office supplies and office infrastructure. For example, a server or desktop product may require specific software and we should include that information on a record. A #2 pencil has no required software.	Anthony	3		Record Types: Created two record types: Office Supplies and Office Infrastructure. Assigned these record types to the appropriate profiles to ensure correct usage. Page Layouts: Office Infrastructure Layout: Included a multi-picklist field to enable users to select multiple software options necessary for the product. Office Supplies Layout: Excluded the multi-picklist field, as these products do not require software. Validation Rule: Added a validation rule to enforce that the Office Supply record type cannot be assigned any software. This rule prevents the assignment of software to office supplies even if attempted through Apex code.		
Process Automation > Process Builder	Ensure that when a technician is assigned to a work order they automatically gain access to the work order record.	Jay	6		Complete **Evgeny's comment - unsure what to retrieve**		
Profiles/Permission Sets or Hierarchy	Only sales team members and their managers may view an Account's customer success stories. Managers should be the only ones with permission to delete customer success stories.	Jay	3				
Validation Rule or Dependent Lookup	When a Work Order is saved into the database, it should contain information about the related Account and the service techician. Ensure that it is not possible for the user to select someone who is not a service technician	Anyone	3				
Process Automation and Logic Max Points: 30							
Flow	FBC would like to enforce an environmental fee on all Orders with a type of "Paper". Ensure that these Orders *always* have this environmental fee on them and this fee is not added to any Orders that do not have a "Paper" type. The fee is a flat $50 and should never be more.	Richard	12		Made a flow that essentially gets all of the information necessary to identify if an order is of Paper type, and if it is, it automatically adds the "Environmental Fee" product to the order upon order creation.		
Apex Trigger	Customer Success Stories are given a rating from 1-5 that allows for decimal places. FBC would like to know how these stories rank against stories associated with the same Account with a field on the Customer Success Story. Ensure this rank is always up to date. For example, if there are 5 Customer Success Stories on the Edge Communications Account, each of those 5 should be ranked against the other 5. Ties should have the same rank.	Anthony	12				
Apex Trigger	Whenever a customer Contact record is saved or updated, we should check that the email field is similar to all other Contacts for that related Account. If there is no similarity, then we should stop the record from being saved. For example, if we have a "john.doe@revature.com", then all records with emails ending in "@revature.net" shouldn't be saved.	Arashdeep	6		Complete. Got Account Ids associated to the Contact. Loop through Contacts and the Contacts associated with Account to check if the domain of email is the same. Create test cases to test implementation.	Classes: ContactHelper ContactHelper_Test Triggers: ContactTrigger	
Flow or Apex/ Consider junction object	Each hardware Product, such as a server, should have a Product Specialist associated with it. A product specialist represents a subject matter expert on that particular piece of hardware. Whenever an Order comes in that includes this product, a pending installation Work Order should be created that relates to the Product Specialist and the Account that placed the order.	Evgeny	6				
Screen Flow and Subflows	FBC would like the customer feedback to be a guided process for the customer so they fill out all the fields properly. Once the feedback has been gathered, the record should be added and any feedback with a rating under a 3 should cause a notification to be sent to the admin. FBC is still working on their customer-facing site, so just make sure it's available internally for the time being.	Jay	6				
Approval process	FBC wants to make sure **Orders are not being activated without approval**. Ensure that an Order goes through an approval by the owner's manager before being activated. Ideally they would like submitting for approval to be easy; such as a button on the order page.	Richard	6				
Flow	FBC wants to make sure Opportunities are being closed on time. Ensure that the owner of an Opportunity gets a task to close the Opportunity if it's not closed a week before the close date.	Anthony	6		Flow Trigger: The flow is set to trigger when an Opportunity is created or updated. Entry Conditions: StageName is not equal to "Closed Won" and StageName is not equal to "Closed Lost". Scheduled Path: The flow includes a scheduled path set to execute 7 days before the Opportunity's Close Date. Decision Node: At the scheduled time, the flow checks if the Opportunity's StageName is still not "Closed Won" or "Closed Lost". If the Opportunity has been closed (either "Closed Won" or "Closed Lost"), the flow terminates and does not create a task. Task Creation: If the Opportunity is still open, the flow creates a Task record assigned to the Opportunity owner. This Task serves as a reminder to close the Opportunity. Testing: Created multiple test Opportunity records with Close Dates on the 8th and 9th of the month to verify that tasks are generated correctly. Additional tests ensure that tasks are not created if the Opportunity is closed before the 7-day threshold.	Flow: Opportunity Task Reminder	
Apex Trigger	Service Technicians are both **Users and Contacts**. Create a profile for Service Technicians and ensure that whenever a User with this profile is created, a Contact is automatically created for them and associated with the User record. If this profile changes, the Contact record should be deleted. Prevent Users without this profile from being associated to a Contact.	Jason	12		Completed before insert and before update trigger. In the trigger query for 'service technician' profile. check for if user exists and is updating to a tech, or is a new user then create contact. Also check for user who is going from tech to non-tech, then delete that contact.		
User Interface Max Points: 25							

Visualforce	Create a Visualforce page for viewing/creating/deleting both customer Contacts and service technician Contacts. The app should have cohesion, allowing us to navigate to other pages within the app.	Evgeny	6		Complete along with test class for the custom controller. Includes ContactManager.vfp page, ContactManagerController.apxc and ContactManagerControllerTest.apxc classes.		
Visualforce	Create a Visualforce page for viewing/creating/deleting Orders and managing which Products are in the Order. The app should have cohesion, allowing us to navigate to other pages within the app.	Richard	12				
Visualforce	Create a Visualforce page for viewing/creating/deleting Accounts and Leads. The app should have cohesion, allowing us to navigate to other pages within the app.	Jay	6		Developed a Visualforce page by creating a controller that handles all the queries and sends them to the view page.	Classes: AccountLeadManagerController Pages: AccountleadManager	
Visualforce	Create a Visualforce page for viewing/creating/deleting Work Orders. The app should have cohesion, allowing us to navigate to other pages within the app.	Arashdeep	6		Completed. Wrote function to create and delete work orders in controller. Created a page with a custom controller with navbar and table	Classes: WorkOrderManagerController WorkOrderManagerControllerTest Pages: WorkOrderManager	
Visualforce	Create a Visualforce page for viewing/creating/deleting Opportunities. The app should have cohesion, allowing us to navigate to other pages within the app.	Anthony	6		Visualforce Page Setup: Created a Visualforce page named OpportunityManagerPage to handle the viewing, creation, and deletion of Opportunities. Controller Logic: Used OpportunityManagerController to manage the business logic for the page. The controller includes methods for querying existing Opportunities, creating new ones, and deleting records. Form for Creating Opportunities: Added inputField tags within a form to capture user input for new Opportunity records. Created a commandButton labeled "Create Opportunity," which invokes the createOpportunity action in the controller. Displaying Opportunities: Used a pageBlockTable to list existing Opportunities. Data is retrieved using an SOQL query in the OpportunityManagerController. Deletion of	Classes: OpportunityManagerController Pages: OpportunityManager	 
Skip	FBC has a number of helpful external links that they would like to display to their internal users. This list changes occasionally, so instead of giving a fixed list of links, they would like to be able to create new external link entries and then have all the links displayed on their own app page. They don't just want the url displayed though, they would like to show the user a preview of the page from within the app itself with a short description below it that they will write when creating new external link entries. Since these sites can change their look, they won't supply the image preview.		12				
Tab and page	Users should be able to easily find and use our Visualforce application within our Salesforce instance.	Jason	2		Completed. Went to tabs in the setup menu created a new VF tab and linked it to a VF page		
Profiles/Visualforce	This use case requires the Contact Visualforce page feature request. Ensure that only HR team members may view/create/delete service technicians.	Richard	4				
Visualforce	FBC's logo should surface on each of our pages within the Visualforce application.	Anthony	2		Static Resource for Logo: Uploaded the FBC logo image as a static resource in Salesforce. Named the static resource appropriately (e.g., FBC_Logo) for easy reference in Visualforce pages. Visualforce Page Updates: Edited all Visualforce pages to include the FBC logo. Ensured the logo appears consistently on each page by using a common header component. Navigation Bar: Refined the navigation bar design using CSS to maintain a cohesive look and feel across the application. Incorporated	CSS: GroupStyles	
Testing, Debugging, and Deployment Max Points: 22							
Apex Unit Testing	Any Apex code written must have unit tests reaching 75% code coverage minimum.	**MANDATORY, ALL ASSOCIATES MUST ASSIST IN THIS USE CASE**	7		Complete.		
VSCode connected to a Playground, upload .csv file through Data Loader	Demonstrate your understanding of deployment with Vscode by deploying your project configurations to a fresh org as part of the live demonstration. Further, you should populate some sample Contact record data into the org using your knowledge of data management tools.	Evgeny	5		Complete.		
	In a section of your presentation, analyze the pros, cons and situations where you would deploy your project using SFDX tools, Managed Packages, Unmanaged Packages, and Change Sets.	Anyone	5				
	In the live demonstration, walk through a debug log that has been generated during the demo. You should be able to discuss each component of the log, log categories, and log levels.	Anyone	5				
	Developers must anticipate issues and handle these errors accordingly. Any Apex code must handle possible exceptions gracefully. If in a Visualforce controller or extension, display the errors on the Visualforce page.	**MANDATORY, ALL ASSOCIATES MUST ASSIST IN THIS USE CASE**	5		Complete.		

Code freeze by Saturday. Work on presentation Sunday.

Presentation: Each associate will take part in a presentation that covers all of the use cases completed as a showcase. Make sure to cover every use case and how it was completed. Note that some use cases are better suited to a live demo than a powerpoint presentation.

You are responsible for presenting a single user epic; pick a particular user of the project and present the use cases that were completed that impact that user. Not every use case is part of the epic for a given user, so these fall into a general epic for the business a whole (i.e. The use case to send an email to hot-rated accounts at the end of every month is not part of a user epic, so it falls into the epic for the business as a whole).

Scoring:
Your grade will be divided into 4 categories: Group, Individual, Presentation, and Agile. The point

Advice:
- Make sure to work as a team. No one person can complete the project alone, but by working together you will better understand what each member is doing.
- Follow these general steps to solve a problem:
  1. Try and solve it yourself.
  2. Search online for the answer; put emphasis on going to the official Salesforce documentation first.
  3. Reach out to your team for assistance. Explain your problem clearly, including what you are trying to do and what you have attempted so far.
  4. If after an hour or so your team can still not solve the problem, reach out to your trainer for assistance.
- Get started early. You may find that you do not know enough information for some use cases until later, but there will always be some work that you can complete.
- Some solutions to use cases will require you to do outside research, since we will not cover them in class. Your analysis step should reveal which use cases these are.
- You will each need to work on your own org to do development. Use the SFDX CLI to pull your changes into a VS Code project and use Git source control. Make sure to commit often and push your changes to GitHub so your team can retrieve your changes.

Point totals of currently complete or working on, excluding shared points:

Jason - 17
Richard - 15
Anthony - 17
Arash - 15
Evgeny - 17
Jay - 12

Jay has one use case remaining to pick in the last category.

[Salesforce Documentation]

In my opinion plan going forward should be:

-Evgeny and Jay pick up two 6-point Visualforce use cases and two use cases in the last category.
-Jason and Richard pick up the two two-point use cases at the bottom of Visualforce which have to be done.
-Arash gets there with a 6-point Visualforce use case.
-Anthony can decide whether he wants the 4-point use case in Visualforce or can trade with Evgeny or Jay in the last category.
-In excess of basic parity at 16-17 use case points help others out or do what you want at your discretion.