Scope

Our Team has been tasked with designing a proof of concept app, capable of predicting the prevalent weather conditions for any location, as determined by the user, on any specified day by using historical weather data thus allowing farmers to plan suitable crops to grow and effective farming methodology to maximize yield

Utilizing the corresponding weather and temperature data for the location and date specified by the user, the app should be capable of predicting whether a particular strain of crop will produce a high or low yield, assuming the weather conditions remain constant throughout the entirety of the season.

Additionally, the app should also present the user with relevant crop information, based on temperature information (for a specified date), such as the predicted state of yield (high/low), safe temperature range and the number of days the crop will survive based on forecasted conditions.

Moreover, since the purpose of this app is to aid farmers in deciding the most sustainable, high-yield crops to grow based on predicted weather conditions, it only makes sense that the most up-to-date weather data be at the disposable of the app to ensure accurate crop yield predictions. However this App will not be using live updated weather forecasting data and instead utilize historical temperature data. This is a substantial failure of the app and is indicative of a significant improvement that future development teams can bring to the app via updates.

Henceforth, the present version of the app will not take into account the impact of climate change around the world thereby reducing the accuracy of forecasted weather and temperature data and in some circumstances result in an incorrect forecast.

As mentioned before, the fundamental purpose of this app is to serve as a proof of concept that farmers are able determine the state of output crop yield for any given crop based on temperature data for a specified location and date, assuming the weather conditions remain constant throughout the entirety of the season.

We assume the weather conditions to be constant for the entire season to ensure uniformity in temperature data since the “survival days” of crops is dependent on the temperature of that specified day, and if the temperature varies significantly over the following days, each of the days to follow will output a different value for the number of days the crop is expected to survive. As such a situation would complicate plans for farmers, the weather conditions are assumed to be constant for the entire season.

The app should also aim to provide the user with a high degree of control over presented information and in that spirit allow the user to search for a specific date then proceed to present him/her with relevant crop yield data for specified date.

The crop data available to the app is based on growth in Australian seasons and therefore Australia’s seasonal timeline is extrapolated for all other countries as well. This allows the app to determine which crops are in season in a specific location based on the underlying assumption that the prevalent season in the chosen location corresponds to the season in Australia in that time of year. (eg. When Australia is in spring, all other locations worldwide experience spring)
Consequently this would also confine the use of this app to the boundaries of Australia.

The app requires the user to login daily as there is no notification system in the present version of the app. Therefore the user will not be alerted of any sudden change in temperature which may cause the farmer to lose out on a significant yield.

TITLE	DESCRIPTION	DUE DATE
1.0 Location and Crop Lists	**Full list of location and crops will be populated on Main Page.**	8/24/18
1.1 Add crop button	A button located on the title bar that opens the Add Crop page.	
1.2 Add Location button	A button located in the title bar that opens the Add Location page.	
1.3 Delete Crop	A button that removes that crop from the list on the main page and updates local storage.	
1.4 View Location	When the user taps on a location, they should be taken to the view location page for that location	
1.5 Delete Location	A button in the view location page which allows the user to remove a location; on tapping this the location should be removed from main page and local storage and the user returned to the index page.	
2.0 Crop Page	**Where new crops are created and any crops created will appear on the main index page.**	8/27/18
2.1 Adding new crops	User is able to create a new crop, by providing all relevant information, and create an instance of the crop class. This instance should be saved to the local storage. Once a crop has been saved the user should be returned to the index page where their new crop should now appear in the list	
2.2 Crop class	This class should include :- - Name of the crop - Seasons in which the crop grows - The safe temperature range for a good yield - the number of degrees beyond the safe range which can become dangerous to the crop (Low Yield Offset) - the number of degrees beyond the safe range which can become dangerous to the crop (Tolerance)	
3.0 Location Pages	**Where new locations are added, Should show location information with historical temperature information and estimated impact on the crops**	8/31/18
3.1 Add Location Page	Allow user to enter location by Suburb and Country using MapQuest's geocoding API and the app should display the given address as an annotation on the map. Entered location should be added to the main list and the local storage updated.	

3.2 Select a location	Users will be directed here after tapping on their selected location on the "Location and crop lists page". On this page they should see a map view clearly showing the location itself, as well as a nickname if one was given. This page also shows weather details for a given date and allow the user to remove the location if desired.	
3.3 Getting temperature data from DarkSky API	The user should be able to choose a date on the view location page (only available for the last 12 months from today) and if it is valid this date should be shown on the view location page. The page should then be update to should the high and low temperature of that day.	
3.4 LocationWeatherCache class	The Class represents historical data held persistently by the app (helps to reduce waiting time when getting data from the Dark Sky API).	
3.5 Getting data from LocationWeatherCache	Since the API allows a limited number of calls and can take some time to return with data, we want to save results as we get them. Using your LocationWeatherCache class you should store weather information and keep this up to date in local storage. Now you will need to check whether that information is in your cache; if the data is in your cache you should retrieve that instead and only if the data isn't in the cache should you make an API request from Dark Sky.	
3.6 Checking which crops are in season	App should determine which of the crops stored in local storage are in season and display these crops along with their safe temperature range in this page.	
3.7 Estimating effect of weather on crop	App will check the highest and lowest temperature of that day against the crop requirements and determine yield, offset range and survival days.	
3.8 Displaying relevant crops and its effects	Update the view page to display the state for each crop in season, assuming that the temperature is consistent throughout the season.	
4.0 Important Considerations	App should store and retrieve data from local storage	
4.1 Storing and retrieving from local storage	Forecasts and Crop data should be saved in local storage and retrieved from local storage when required to improve efficiency and speed of the app.	

Finalisation of code	**- Adding comments to ensure ease of understanding** **- White box testing by the members**	9/6/18

Technical Documentation		
Project Management Plan.	This is internal documentation detailing: - Project aim and scope - Team members, and their responsibilities - Project operations - team meetings, minutes, handling communication of changes to the code, processes - Any other information you would want to give to a new team member	
User guide	Includes the following: - Introduction and its purpose - Features in the App - Instructions on how to use various features of the app - Known bugs and limitations of the app - Sreenshots of the app	9/6/18
Class diagram	A type of static structure diagram that shows the full set of classes used in the code, their operations and relationships among objects.	8/31/18
Presentation	Client Presentation with full team	9/9/18