Sprint 2 Planning Document
TEAM 5
WPEAR

Stephen Harrell, Lala Vaishno De, Mengxue Luo, Dhairya Doshi

1. Sprint Overview

This sprint will be focused on expanding on our core functionality. A key functionality this sprint will be to get interpolation working. Beyond that we are taking each of our functionalities and extending them forward with features like moving visualizations, conversion into common data formats and data archival. By the end of the sprint, we will have a command-line tool that can download files for forecast and observation data, compare forecasts with an observation and be able to visualize a forecast, observation and the temperature difference between forecasts and an observation.

Scrum Master: Stephen Harrell
Scrum Meeting Time: Monday at 3:00pm and Thursday at 4:00pm
Risks/Challenges:

- Downloading data files from Real-Time Mesoscale Analysis (RTMA) System Products
- Interpolation of GRIB messages
- Writing out the GRIB2 file format
- Learning to perform moving visualizations from a GRIB2 format

2. Current Sprint Details

1) **User Story:** As a user, I would like to obtain latest observation data from the given sources.

Task Description	Estimated Hours	Owner
Configure downloader to download from new source (RTMA)	1	Lala

Data Source: http://www.nco.ncep.noaa.gov/pmb/products/rtma/

Acceptance Criteria:

1. Given the filename, the file is downloaded and the location of the downloaded file is returned.
2. Allow listing directories from RTMA products

2) **User Story:** As a user, I would like to have the data interpolated before evaluation for the visualizations.

Task Description	Estimated Hours	Owner
Be able to interpolate a forecast data file to a grid size of 1 sq. km.	15	Stephen
Be able to interpolate an observation data file to a grid size of 1 sq. km.	5	Stephen
Test to ensure the interpolated objects for forecasts and observations can be compared. (compatible with each other)	5	Stephen

Acceptance Criteria:

1. Interpolate the data values of a given message object belonging to a forecast file to the required grid size.
2. Interpolate the data values of a given message object belonging to an observation file to the required grid size.

3. The interpolated objects from a forecast and an observation file can be compared. Passes test cases to prove their compatibility.

3) **User Story:** As a user, I would like to choose between different variables (for instance dewpoint or temperature) for a particular visualization.

Task Description	Estimated Hours	Owner
Be able to extract a message from a grib2 file using the name of the variable	15	Lala
Be able to extract multiple messages from a grib2 file using the name of variables	3	Lala
Write the extracted messages into a new file with a given filename	2	Lala

Acceptance Criteria:

1. 1 message is extracted for a given single variable and written into its own individual file.
2. 10 messages are extracted for a variable list containing 10 messages and written into its own individual file.

4) **User Story:** As a developer, I would like observation data to be subsetted for given region.

Task Description	Estimated Hours	Owner
Examine the format of the observation data files (RTMA products)	5	Lala
Find a way to subset for a particular region	5	Lala
Store the subsetted region for the file in its own file	1	Lala

Acceptance Criteria:

1. Given a pair of latitudes and longitudes and a message, subsetted region for a file is returned in a new file.

5) **User Story:** As a developer, I would like forecasted data to be subsetted for given region.

Task Description	Estimated Hours	Owner
Examine the format of the observation data files (HRRR products)	5	Lala
Find a way to subset for a particular region	5	Lala
Store the subsetted region for the file in its own file	1	Lala

Acceptance Criteria:

1. Given a pair of latitudes and longitudes and a message, subsetted region for a file is returned in a new file.

6) **User Story:** As a developer, I would like to be able to calculate the mean temperature for a given region over multiple forecasts.

Task Description	Estimated Hours	Owner
Create a method that takes in multiple files from the WPEARController	4	Dhairya
Run through each location point in the grib2 file and calculate the average temperature	6	Dhairya
Return a grib message that contains the average	4	Dhairya

Acceptance Criteria:

1. Given that we have the GRIB files, we should be able to add up the temperatures at that location over multiple files.

2. Given that we can add up the temperatures for each location, we should then be able to average that over the number of files we are provided.
3. Given that we can calculate the average, we should be able to return the results as a GRIB message.

7) **User Story:** As a user, I would like to view a static visualization of the comparison done using root mean square difference.

Task Description	Estimated Hours	Owner
Generate a scale and reference colors to display the difference in the form of a visualization	3	Mengxue
Create the visualizations for the difference and display it on a map of the region.	3	Mengxue

Acceptance Criteria:

1. Given that I can calculate the root mean square difference, I expect to be able to generate a heatmap to display the difference

8) **User Story:** As a user, I would like to view a moving heatmap of the forecasted weather. (daily version)

Task Description	Estimated Hours	Owner
Get data from converted files for a specific date (so there will be 24 files for a day)	4	Mengxue
Visualize the data dynamically on map over the chosen domain	10	Mengxue
Generated the corresponding GIF file	3	Mengxue

Acceptance Criteria:

1. Given that we are able to access the local converted files for a day, we could get the needed data for visualization.
2. Given the dynamic visualization is implemented properly, we should be able to generate the correct GIF file as output.

9) **User Story:** As a user, I would like to view a moving heatmap of the observed weather. (daily version)

Task Description	Estimated Hours	Owner
Get data from converted files for a specific date (so there will be 24 files for a day)	4	Mengxue
Visualize the data dynamically on map over the chosen domain	10	Mengxue
Generated the corresponding GIF file	3	Mengxue

Acceptance Criteria:

1. Given that we are able to access the local converted files for a day, we could get the needed data for visualization.
2. Given the dynamic visualization is implemented properly, we should be able to generate the correct GIF file as output.

10) **User Story:** As a user, I would like to be able to store the models for each observation and forecast in separate files.

Task Description	Estimated Hours	Owner
Be able to store the converted observation and forecast files.	3	Stephen
Determine a file organization system to archive the converted files.	3	Stephen
Be able to search the file system archive for previous conversions and retrieve it into an object.	2	Stephen

Acceptance Criteria:

1. Converted observation and forecast files can be saved onto the local file system in the desired location.

2. File organization system developed to handle all converted files.
3. Search and retrieve within the file organization system works successfully.

11) **User Story:** As a developer I would like a class that orchestrates the downloading, conversion, comparison and visualization of the data

Task Description	Estimated Hours	Owner
Figure out right file to download hourly and runs DataDownloader properly to download the file	3	Stephen
Runs DataConverter properly with downloaded files to convert the file in an expected way	3	Stephen
Runs DataComparator properly with files need to be compared to generate comparison output for further data visualization	1	Stephen
Runs DataVisualizer properly with right files to generate wanted data visualization	1	Stephen
Runs WebsiteGenerator properly to render the wanted output on front end	1	Stephen
Handles errors and exceptions happening during the workflow properly	1	Stephen
Organize the local file storage properly and clean up the downloaded large grib file periodically	4	Stephen

Acceptance Criteria:

1. Given that each component is implemented correctly, the WPEARController is able to supervise the working flow by giving instructions to each individual component.
2. Given that each component is implemented correctly, the WPEARController is able to handle running errors and exceptions properly during the process running.
3. Given that WEAPRController is implementedly properly, the local file storage should be organized well for future historical data trends analysis.

12) **User Story:** As a user, I would like to view the results via website.

Task Description	Estimated Hours	Owner
Update the website to include new user input sources like choosing variables that they want to see	5	Dhairya
Implement necessary support to display moving heatmaps (gifs)	5	Dhairya
Update the website to provide additional data and make it more visually appealing	4	Dhairya

Acceptance Criteria:

1. Given that we decide on the variables we will display information for, we should be able to provide the user with a choice of the variable they want to see information about
2. Given that we are able to generate moving heatmaps, we should be able to integrate this with the website
3. Given that all the necessary information is available, we should be able to make the website more presentable and informative for the user.

13) **User Story:** As a user, I would like to be able to see the trend in temperature for a given region (5 x 5 mile around a given point).

Task Description	Estimated Hours	Owner
Create a method that takes in multiple files from the WPEARController along with co-ordinates for a given point	3	Dhairya
Find 4 points that enclose the region (forming an imaginary box around the point)	8	Dhairya
Average the temperatures over the 4 points to calculate the average temperature in the region	4	Dhairya

Create a trend line graph for this that can be displayed on the website.	6	Dhairya

Acceptance Criteria:

1. Given that we have the GRIB files and the co-ordinates for the point, we should be able to find the 4 points in the GRIB file that enclose the region
2. Given that we can find the 4 points, we should be able to calculate the average temperature over the 4 points for every file individually.
3. Given that we can calculate the average for each point, we should be able to create a trend graph to display the temperature trend.

Summary of allocated time:

Team Member	Estimated Total Hours
Stephen	40
Lala	43
Mengxue	40
Dhairya	49
Total Hours Assigned	172 hours

3. Remaining Backlog

(a) Include all the other user stories from your Product Backlog document.

Functional Requirements:

1	As a developer, I would like to be able to calculate the mean of a particular variable over time or over a specific region.
2	As a user, I would like to view a moving visualization of the comparison done using mean. (if time permits)
3	As a user, I would like to view a graph showing the accuracy of a specific variable over a time period (if time permits).
4	As a user, I would like to view a graph showing the accuracy of a specific variable based on how far out have they been forecasted (if time permits).
5	As a user, I would like to see historical trends and visualizations
6	As a user, I would like to see the difference between the results from top accurate weather forecast and the worst accurate weather forecast. (if time permits)
7	As a user, I would like to see which model is better at predicting extreme weather. (if time permits)
8	As a user, I would like to see if there exist visible trends in the data that help better predict weather disturbances. (if time permits)

Non-Functional Requirements:

1. **Web Enabled:** Visualizations must be available over the web. Webpages must be autogenerated based on the visualizations that are to be displayed using type of visualization, date/time, location and variable as parameters.

2. **Intermediate Data Archival:** Interpolated intermediate data must be available historically for reanalysis of visualizations. Archived data must be clearly marked with location, time/date and variable parameters. Archived data must be available via the web and linked to the visualizations that are created from it.

3. **Operational Weather:** Every hour this tool should retrieve the observations and forecasts, convert them to a common grid spacing and format and create visualizations based on an evaluation of the observation and forecast for specific points in time.

4. **Web access must be fast:** Webpages should be static HTML/CSS and not include any server side programing.

5. **Modular:** Code must be modularized in a way that adding new types of observations and forecasts is trivial.