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 MRMS MultiI-Radar/Multi-Sensor 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 (MRMS)	2	Lala

Data Source: http://www.nssl.noaa.gov/projects/mrms/

Acceptance Criteria:

1. Given the filename, the file is downloaded and the location of the downloaded file is returned.

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 reflectivity or temperature) for a particular visualization.

Task Description	Estimated Hours	Owner
Allow usage of flags to determine the variable for converting forecast/observation files.	3	Lala
Allow usage of flags to determine the variable for performing comparison.	3	Dhairya
Allow usage of flags to determine the variable required for producing the visualization.	3	Mengxue

Acceptance Criteria:

1. Standard set of flags are defined.
2. Standard set of flags are used and recognized by DataConverter, DataVisualizer and for comparison.

4) **User Story:** As a developer, I would like for observation data to be in a common format (grid spacing and variables)

Task Description	Estimated Hours	Owner
Investigate the grid size and variables that are present in the observation file	3	Lala
Decide on a common grid size and variable nomenclature for the observation and forecast files	4	Lala
Convert the values in the observation file to the common format	10	Lala
Store the results into a new file that can be used by the other functions	5	Lala

Acceptance Criteria:

1. Given that we can get the observation files, we should explore the contents of the file and understand the variables and grid size provided in the file.
2. Given that we have decided on the common format, we should be able to convert the existing data set from the forecast files into the common format (grid size and variables)
3. Given that we are able to convert the forecast files, we should be able to save the files in the new format for later use.

5) **User Story:** As a developer, I would like for forecast data to be in a common format (grid spacing and variables).

Task Description	Estimated Hours	Owner
Decide on a common grid size and variable nomenclature for the observation and forecast files	4	Lala
Convert the values in the forecast file to the common format	10	Lala
Store the results into a new file that can be used by the other functions	3	Lala

Acceptance Criteria:

1. Given that we have decided on the common format, we should be able to convert the existing data set from the forecast files into the common format (grid size and variables)
2. Given that we are able to convert the forecast files, we should be able to save the files in the new format for later use.

6) **User Story:** As a developer, I would like to be able to calculate the root square mean difference of a particular variable over time or over a specific region.

Task Description	Estimated Hours	Owner
Verify that the observation and forecast data is in the correct format	6	Dhairya
Calculate the RMS values for the data set	8	Dhairya

Compare the RMS values between the forecast and the observed file and return the difference as a GRIB object	8	Dhairya

Acceptance Criteria:

1. Given that we have the files, we should be able to calculate the RMS values for the variables in the file
2. Given that we can calculate the RMS values, we should return a grib object that compares the RMS values and records the difference.

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 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	2	Stephen

previous conversions and retrieve it into an object.		

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	1	Stephen
Runs DataConverter properly with downloaded files to convert the file in an expected way	1	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	1	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
Add feature to allow user to choose comparison type (difference, RMS etc.)	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 other metrics like RMS and mean, we should be able to display visualizations for this to the user.
3. Given that we are able to generate moving heatmaps, we should be able to integrate this with the website
4. Given that all the necessary information is available, we should be able to make the website more presentable and informative for the user.

Summary of allocated time:

Team Member	Estimated Total Hours
Stephen	40
Lala	42
Mengxue	43
Dhairya	44
Total Hours Assigned	129 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.