Name of Project	*A web application for the AI research community in South Africa - AI-Web*
Name of Client	*Deshen Moodley*
Team member 1	*Shreeya Khoosal, KHSSHR001*
Team member 2	*Rauseenah Upadhey, UPDRAU001*
Team member 3	*Kauthar Orrie, ORRKAU001*
Stage:	*2*

Use Case Diagram:

Use Case Narratives

Search Researcher Name
Actor: **AI community of South Africa**

The user searches for researcher by name. In response to this, the system performs a SQL query of the existing dataset. If a match is found, the requested researcher's details are formatted and returned to the user.

Alternative flow of events:
If there is no match for the search an appropriate message communicating this is displayed and prompting user to try again.

Filter Researchers
Actor: **AI community of South Africa**

The user filters an existing list of researchers according to parameters such as their rating or research field. The researcher data is reordered/refined by the system which subsequently displays the new filtered list of researchers.

Alternative flow of events:
Input parameters will be constrained by the user interface and therefore no data validation (necessitating an alternative path) is required.

View trends within AI community
Actor: **AI community of South Africa**

The user requests to view trends in the AI research community data. The system responds to this action by running a predefined set of functions that highlight patterns in the data, such as reporting on the area/institution which has the highest number of researchers. These trends are then displayed in a visually accessible manner such as a bar graph or pie chart.

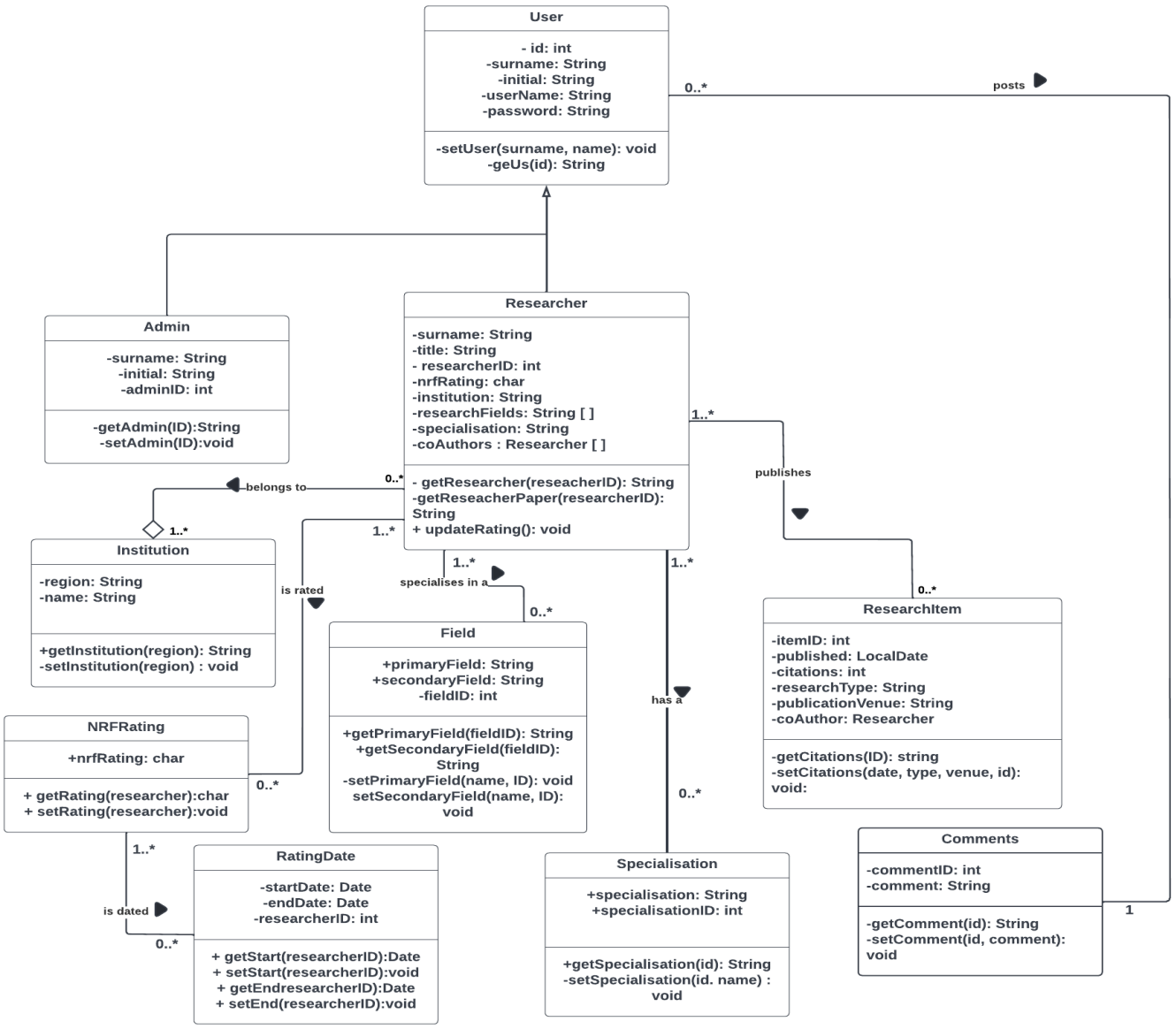
Alternative flow of events:
Input parameters will be constrained by the user interface and therefore no data validation (necessitating an alternative path) is required.

Upload NRF/MAG data
Actor: **Admin**

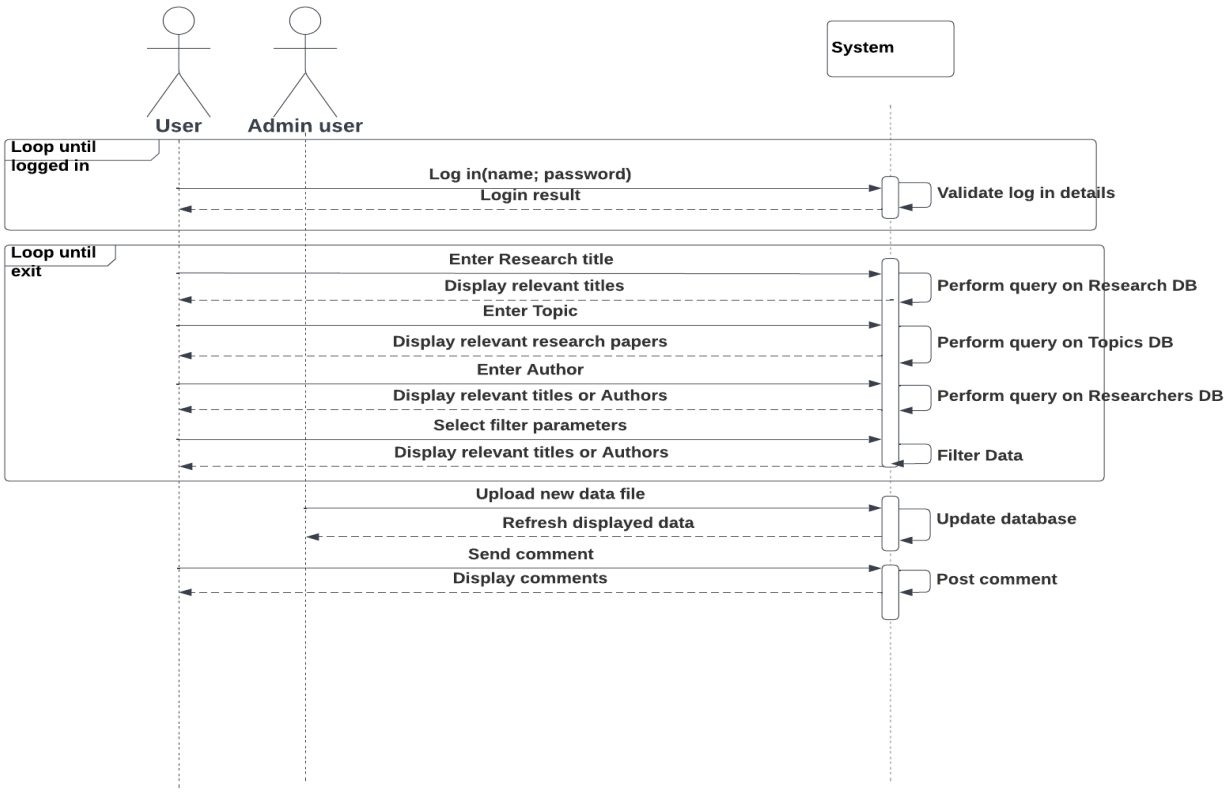
Manual updates to the data can be carried out by superusers/administrators. After logging in, users can initiate an update to the data. The system then updates the data to reflect any changes made to the MAG/NRF data sources since the last system update.

Alternative flow of events: If data uploaded is corrupt or in an incorrect format, the system will display an unsuccessful upload message, prompting the administrator to try again.
Writing a comment Actor: **AI community of South Africa** The user can interact with the web app by writing a comment after reviewing/interacting with the data. The user can type their comment into a comment box and the system will take in the user input and post the comment to the UI.
Alternative flow of events: If the comment is not posted successfully the user will be notified of the unsuccessful posting and will be prompted to try again.

Analysis class diagram:



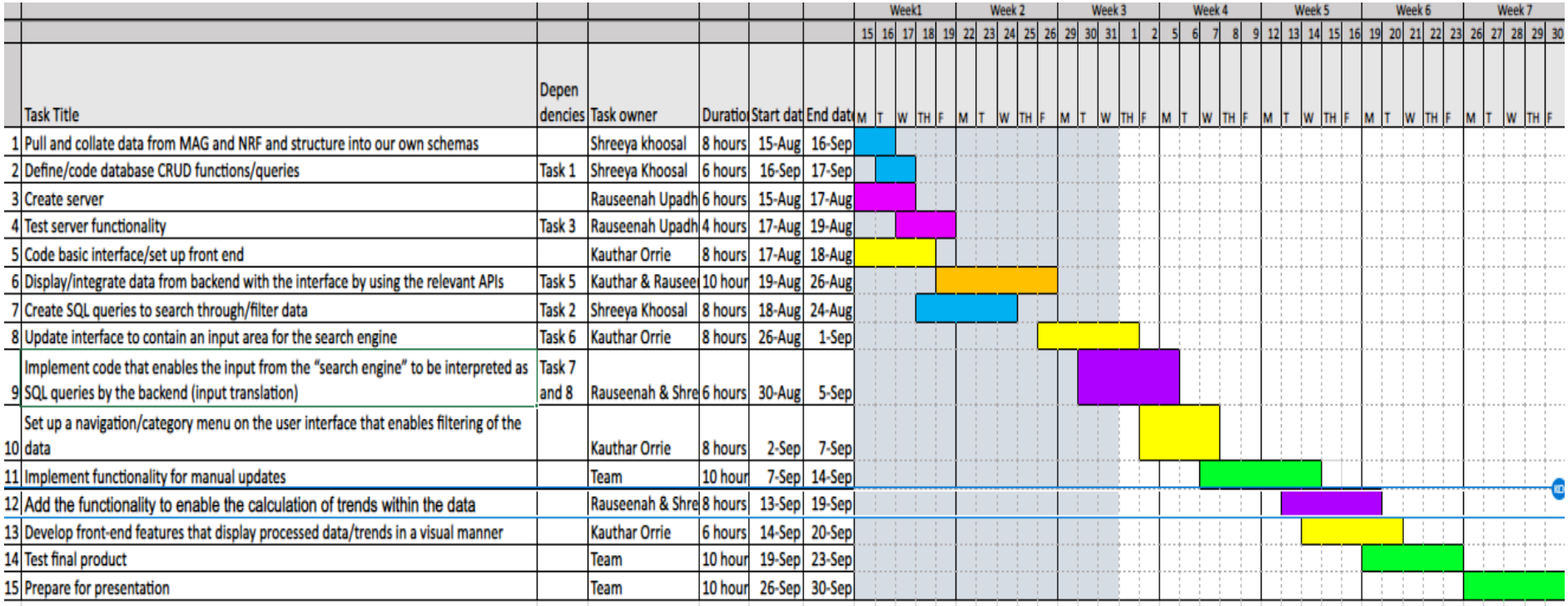
Interaction Diagram:



Project Plan

Please use this link to view the Chart if anything is unclear: [Gantt Chart Stage 2]

	Task Title	Dependencies	Task owner	Duration	Start date	End date
1	Pull and collate data from MAG and NRF and structure into our own schemas		Shreeya khoosal	8 hours	15-Aug	16-Sep
2	Define/code database CRUD functions/queries	Task 1	Shreeya Khoosal	6 hours	16-Sep	17-Sep
3	Create server		Rauseenah Upadh	6 hours	15-Aug	17-Aug
4	Test server functionality	Task 3	Rauseenah Upadh	4 hours	17-Aug	19-Aug
5	Code basic interface/set up front end		Kauthar Orrie	8 hours	17-Aug	18-Aug
6	Display/integrate data from backend with the interface by using the relevant APIs	Task 5	Kauthar & Rausee	10 hour	19-Aug	26-Aug
7	Create SQL queries to search through/filter data	Task 2	Shreeya Khoosal	8 hours	18-Aug	24-Aug
8	Update interface to contain an input area for the search engine	Task 6	Kauthar Orrie	8 hours	26-Aug	1-Sep
9	Implement code that enables the input from the "search engine" to be interpreted as SQL queries by the backend (input translation)	Task 7 and 8	Rauseenah & Shre	6 hours	30-Aug	5-Sep
10	Set up a navigation/category menu on the user interface that enables filtering of the data		Kauthar Orrie	8 hours	2-Sep	7-Sep
11	Implement functionality for manual updates		Team	10 hour	7-Sep	14-Sep
12	Add the functionality to enable the calculation of trends within the data		Rauseenah & Shre	8 hours	13-Sep	19-Sep
13	Develop front-end features that display processed data/trends in a visual manner		Kauthar Orrie	6 hours	14-Sep	20-Sep
14	Test final product		Team	10 hour	19-Sep	23-Sep
15	Prepare for presentation		Team	10 hour	26-Sep	30-Sep



	KEY
1	Team
2	Shreeya khoosal
3	Rauseenah & Shreeya
4	Rauseenah Upadhey
5	Kauthar & Rauseenah
6	Kauthar Orrie

Test Plan

Test Case number	Inputs, behaviour and expected outcomes
1	In order to test the functionality of the search engine feature, the user should enter inputs into the search engine such as the Researcher name. The system is then expected to perform a query on the dataset and return the relevant data. This data (the requested Researcher's profile details) will subsequently be formatted and displayed on the web interface for the user to view. If the correct details have been retrieved and displayed from the database, the test will be regarded as successful.
2	Updates to the data will be tested by clicking the "update NRF rating" button after logging in. The platform should receive their new rating as input, carry out basic data validation tests (to ensure that the format/data type is correct) and perform an update query on the database. A message displaying whether the update was successful will be displayed to the user. If the data has been updated, and the user has been informed on their update status via the interface, then the conditions for this test case will have been met.
3	In order to the test the "view trends or statistics" feature, users should select an option from a list of basic categories such as "number of researchers per institution". The option selected will serve as input to the platform. In response to this trigger, the relevant data should subsequently be retrieved from the database, and formatted in a visual representation, such as a bar graph. If the data has been correctly processed and displayed on the user interface, then the test has been passed.
4	The filtering functionality should be tested by selecting different filter options which are displayed above the research list. If the user selects filter parameters such as a research topic or date of publication, then the platform should process this input by feeding it to the backend, performing a query and returning the relevant data. A filtered list of data will then be displayed on the web interface. If the research list has been successfully re-ordered according to the parameters selected by the user, then the test conditions will have been met.
5	Login functionality should be tested by clicking the login button. The user should be able to provide credentials by entering a username and password into their respective text fields, in order to gain access to the platform. Invalid details should return a detailed error message to the user. The test will be conducted by entering an incorrect password, a username that is not in database, and valid login details. In each case the platform is expected to return feedback to the user regarding the status of their login attempt.