



Hyper Perform Functional Requirements Specification

Repository: <https://github.com/claudioMDS/hyper-perform>

Developers:

Claudio Da Silva *14205892*

Rohan Chhipa *14188377*

Avinash Singh *14043778*

Jason Gordon *14405025*

Updated July 29, 2016

Contents

1	Introduction	3
2	Vision and Objectives	3
2.1	Vision	3
2.2	Objectives	3
3	User Management	4
3.1	Scope	4
3.2	Domain Model	6
4	Integration and Pre-processing	7
4.1	Scope	7
4.2	Domain Model	8
5	Algorithms	9
5.1	Scope	9
5.2	Domain Model	10
6	Reporting	11
6.1	Scope	11
6.2	Domain Model	12
7	Notifications	13
7.1	Scope	13
7.2	Domain Model	14

1 Introduction

Many different tools are available for measuring the quality of products made, but very few tools exist which assess the quality of the people making said products. People play a huge role in a project, and trying to monitor each and every one becomes a tedious task which diverts man power away from other more critical tasks. Whether it be for an end of year evaluation, or attempting to assess the current status of a project, generating a report on a staff member can help keep up productivity, as well as get them any help they need in order to resume quality performance. By ensuring that there is constant quality performance from each individual on a project, one can increase project quality as well as reduce project risks such as loss of an important team member during a critical stage of a project's life-cycle.

2 Vision and Objectives

2.1 Vision

The vision of this project is to create an automated performance management system, which can assess the performance and status of staff members, based on information sourced through various software systems such as card readers, version control systems and such. The system would make use of these external integrations as well as direct contact with the staff members via either web dashboard or mobile phone, to generate reports on the various staff members as well as add elements such as gamification and monitor problems that may be occurring.

2.2 Objectives

The objectives for the Hyper Perform system are:

- To source information from various forms of software systems and transform it into usable data to determine performance.
- To generate real-time reports on staff members in order to evaluate performance based on this data.
- To monitor staff members in order to determine possible causes of work detriment.
- To add a form of gamification to encourage productivity, and discourage slacking and other bad behaviours.

3 User Management

3.1 Scope

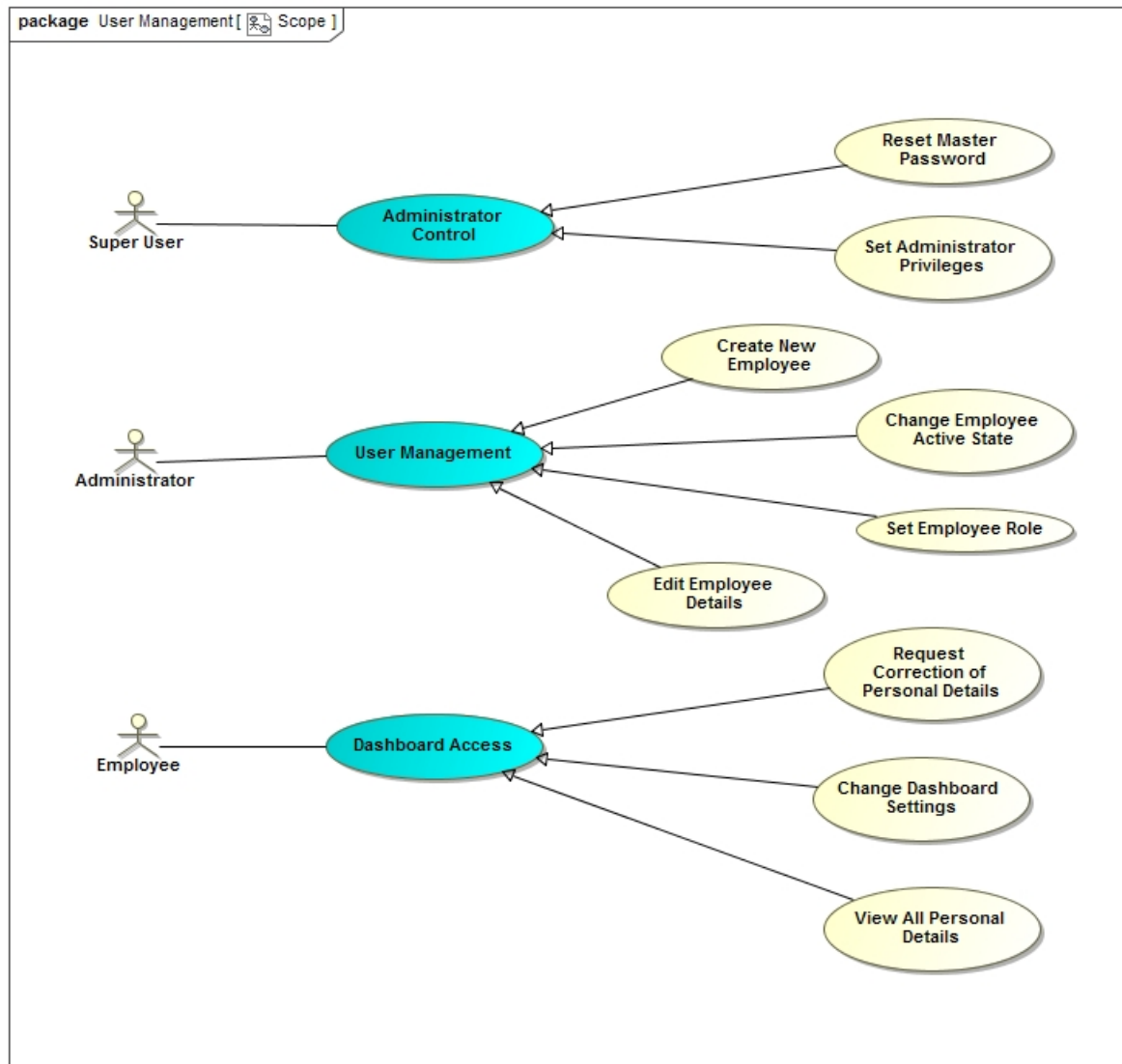


Figure 1: User Management Scope

The scope of the user management module includes:

- Super user control over administrator rights given to individuals.
- Administrators may add employees and manage their details.
- Administrators may add employee roles to employees registered, which will define which algorithms and systems will influence their performance ratings.
- A user may change their dashboard to reflect what they wish to see.
- A user should be able to view all their personal details, and request change immediately if something is wrong.

Note, it is assumed that a super user is not in anyway an employee within the system. It is also assumed that an Administrator is automatically an employee of the system. Administrators may not review themselves or change their own details, such action must be authorised by another administrator.

3.2 Domain Model

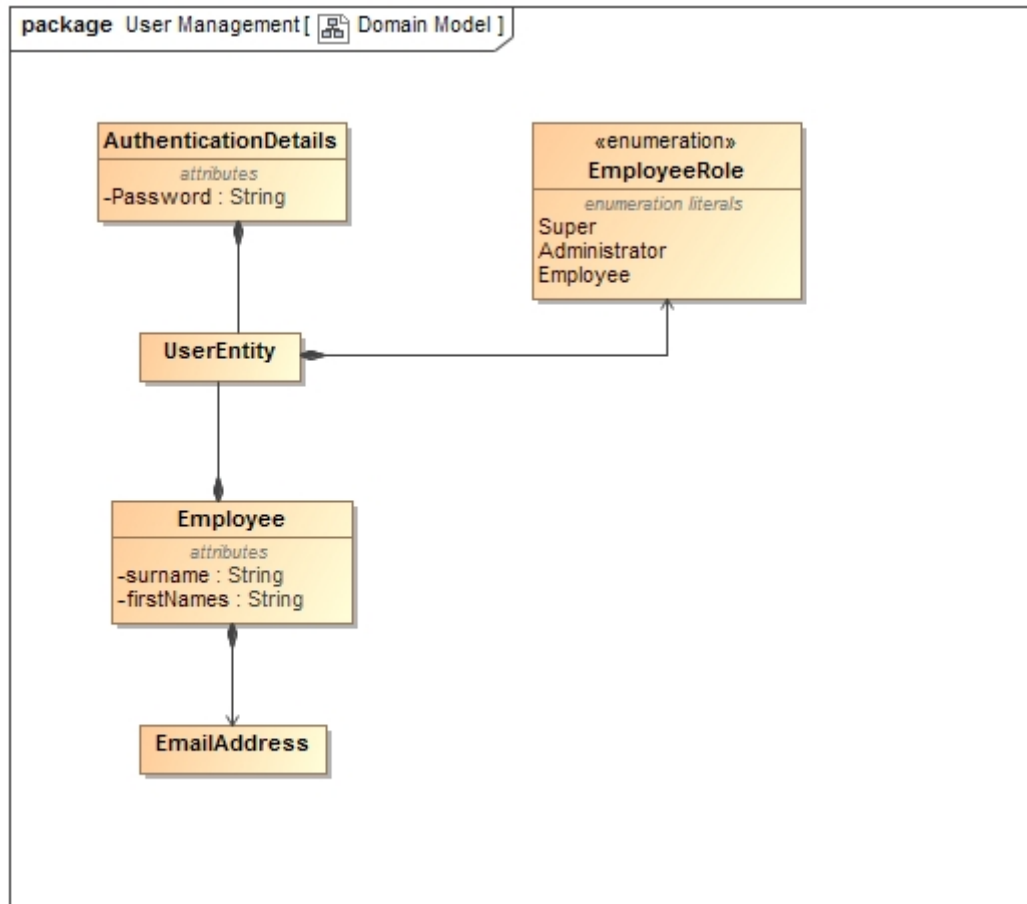


Figure 2: User Management Domain Model

4 Integration and Pre-processing

4.1 Scope

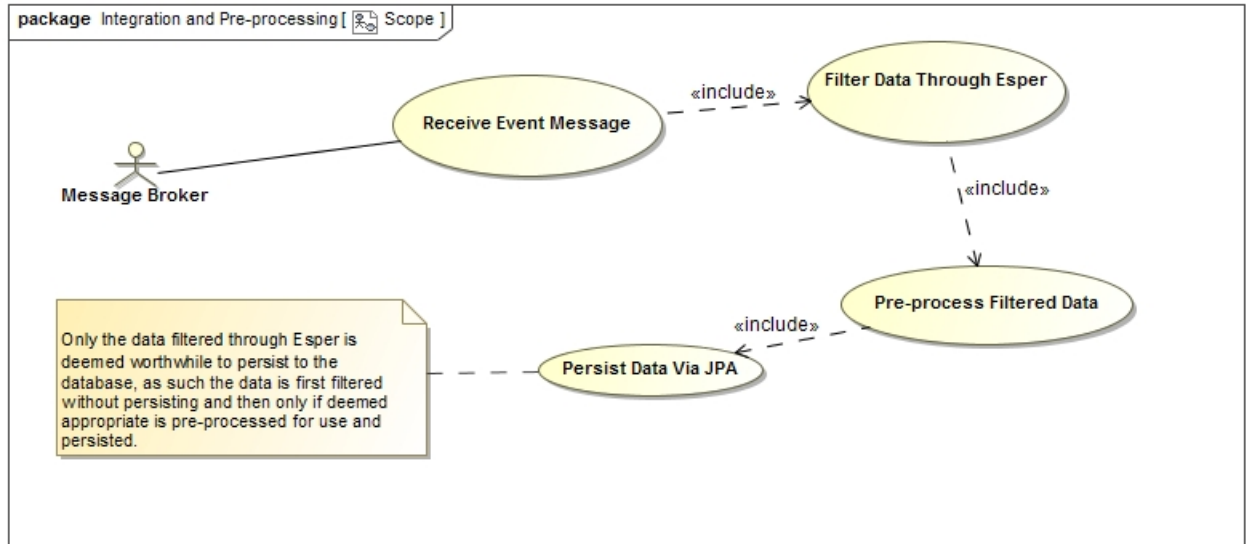


Figure 3: Integration and Pre-processing Scope

The scope of the integration module includes:

- A message broker intercepts the messages and stores them on a message queue.
- Once possible, an Esper query is made in order to filter out the data which is relevant to the system.
- The data that is deemed relevant is then persisted and preprocessed in order to be used by the algorithms module.

4.2 Domain Model

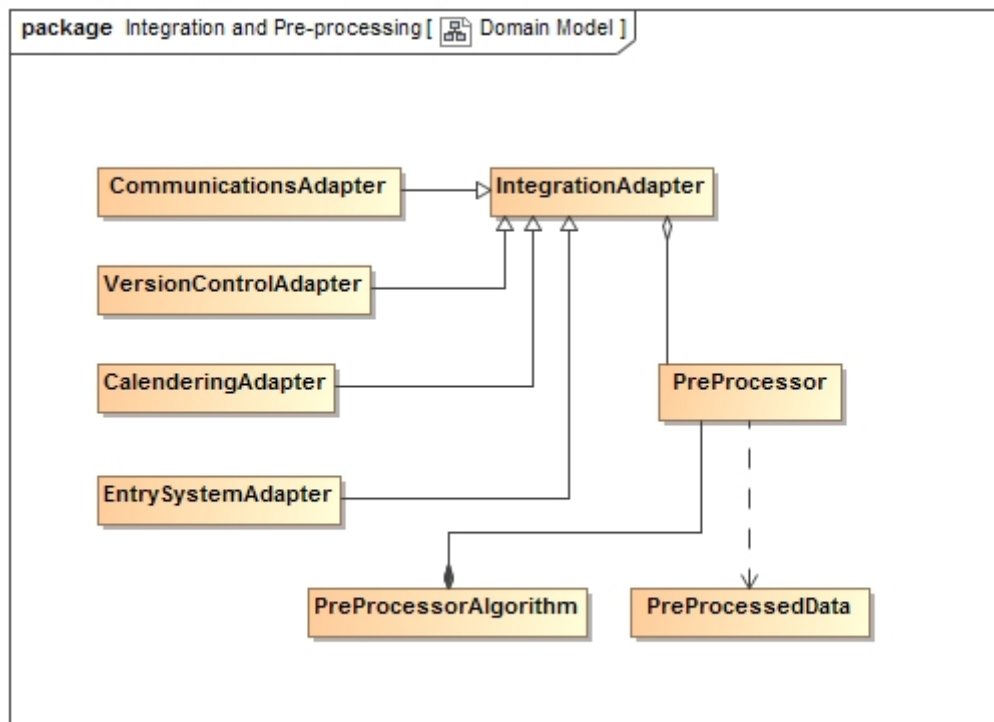


Figure 4: Integration and Pre-processing Domain Model

5 Algorithms

5.1 Scope

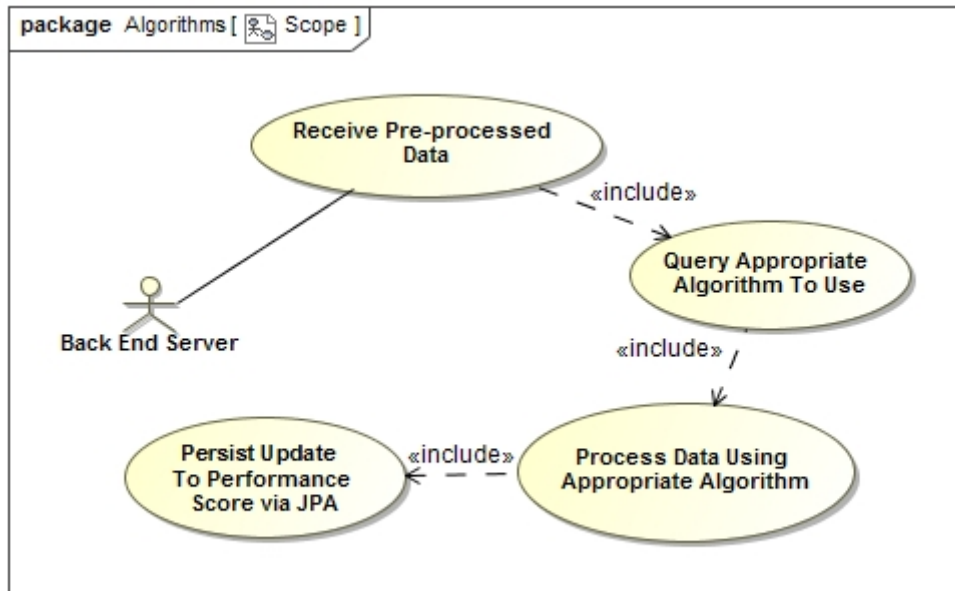


Figure 5: Algorithms Scope

The scope of the algorithms module includes:

- Determining which algorithm "recipe" will be appropriate in the current circumstance.
- Applying said "recipe" to the data provided, and then updating the employee in question's performance score accordingly.

5.2 Domain Model

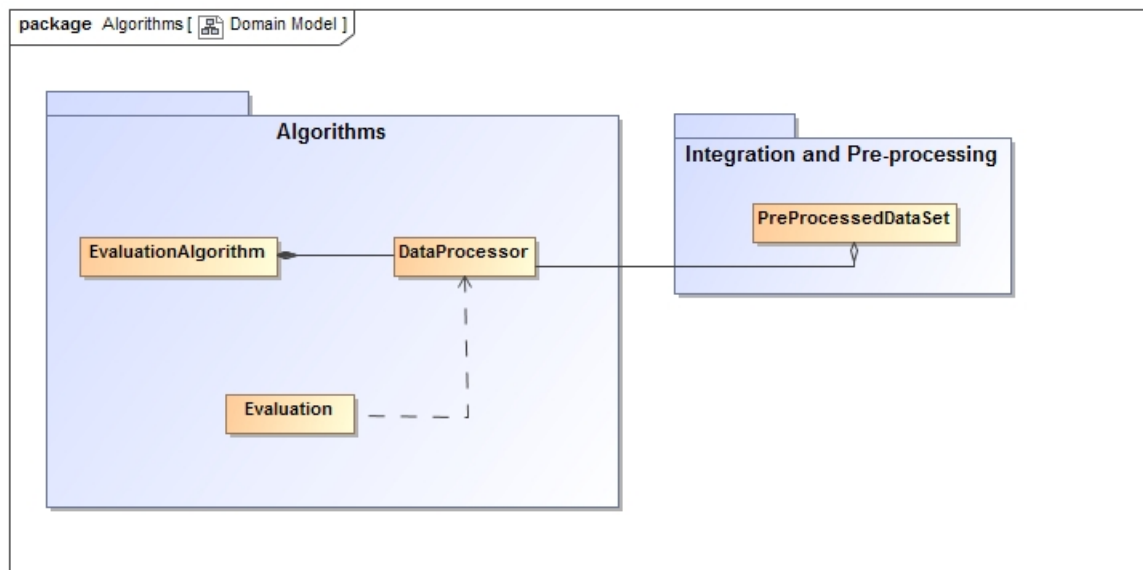


Figure 6: Algorithms Domain Model

6 Reporting

6.1 Scope

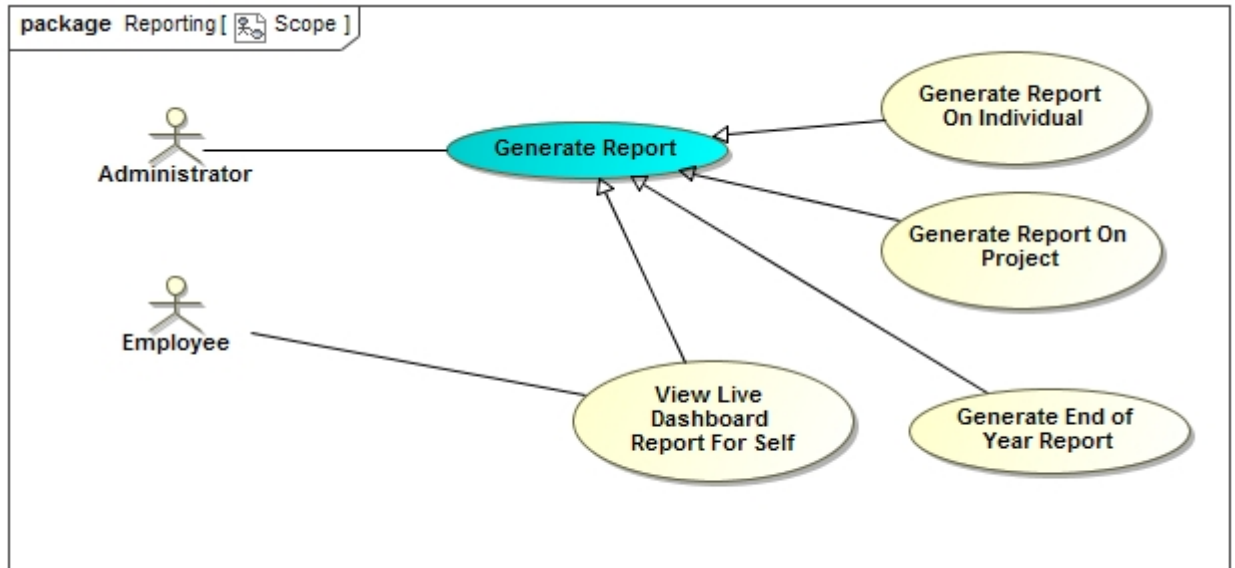


Figure 7: Reporting Scope

The scope of the reporting module includes:

- Administrators such as HR may request reports on individual employee's performance.
- Administrators may also request reports on the current project performance, as well as an end of year final report.
- All users, including administrators whom are users, may view their own dashboard which contains information regarding their current performance and personal details.

6.2 Domain Model

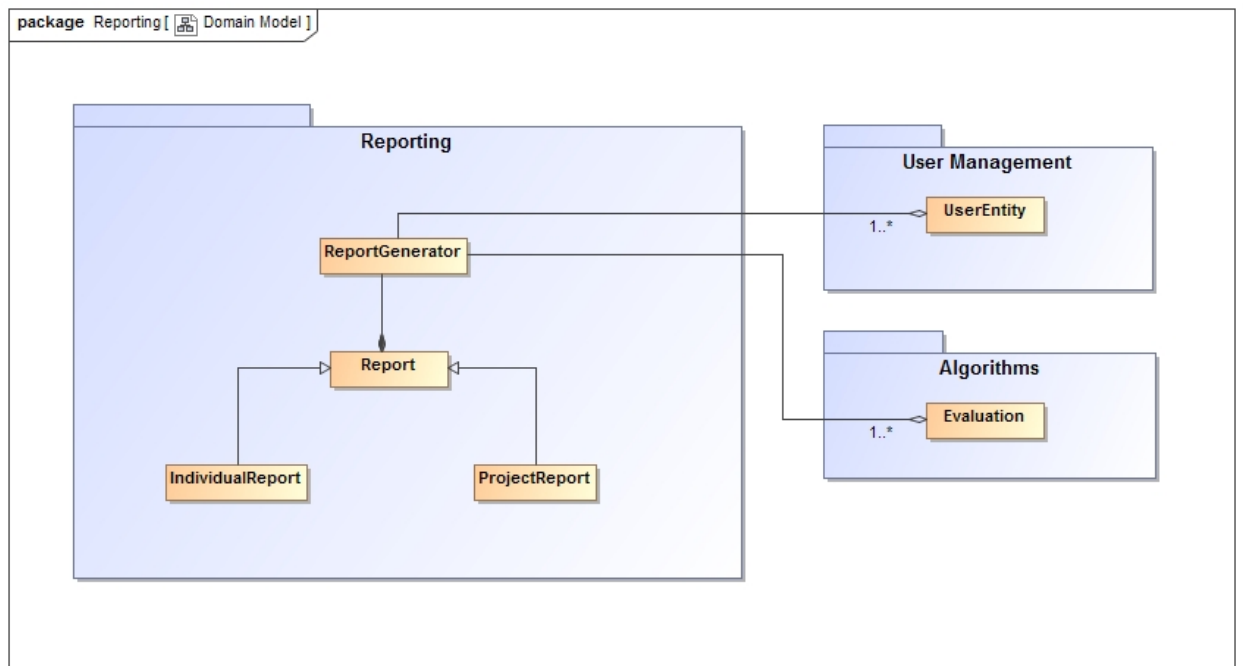


Figure 8: Reporting Domain Model

7 Notifications

7.1 Scope

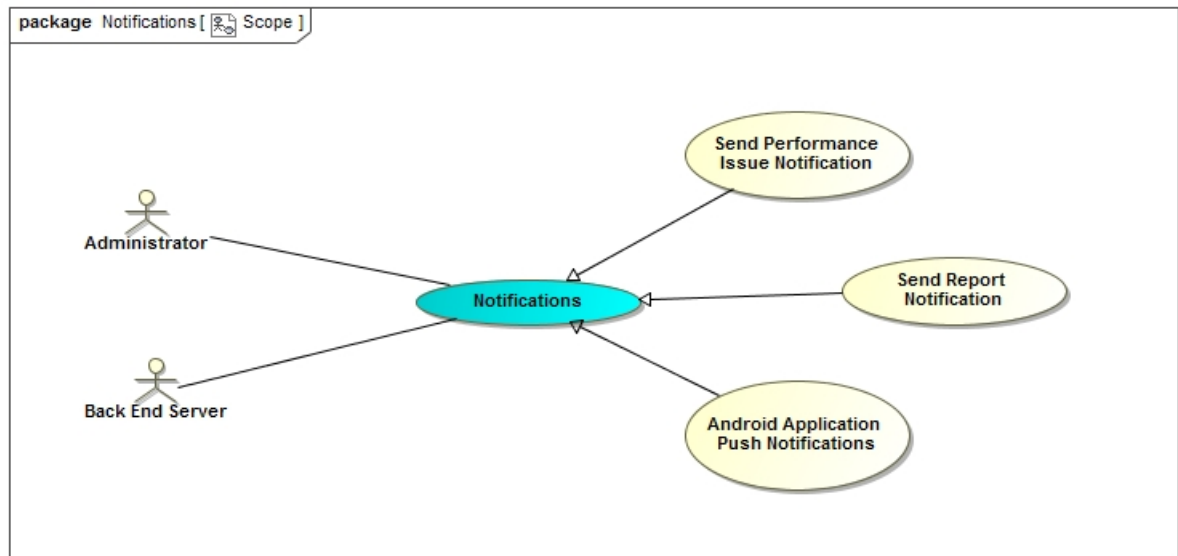


Figure 9: Notifications Scope

The scope of the notifications module includes:

- Sending a notification about a rapid decline in performance.
- Send a notification about a report that an employee requests.
- Push notifications are sent to the Android device to notify of important events or required surveys.

7.2 Domain Model

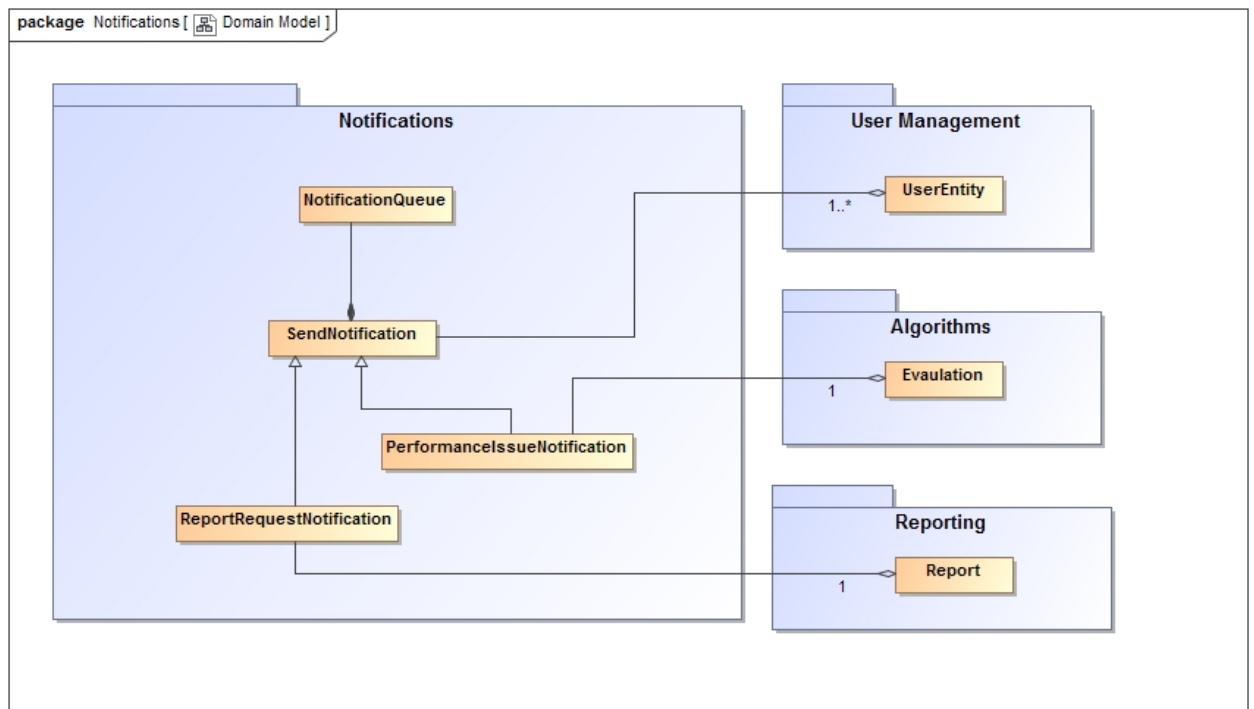


Figure 10: Notifications Domain Model