**Iteration 0**

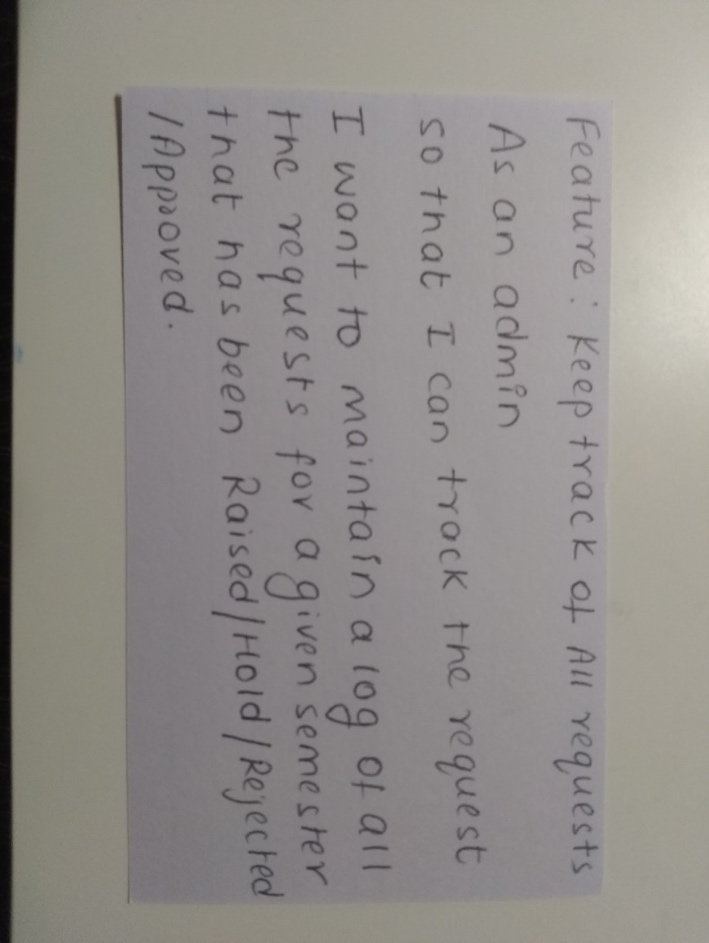
**Team:** ShadowCoders -- Force Request System  **Product Manager:** Niti Jain **Scrum Master:** Shamshu Shahid Raja Mohamed **Other Members:** Abdullah Abdul Kader, Karimi Abhishek Das, Nishant Aditya, Ramesh Ghimire, Sulav Adhikari

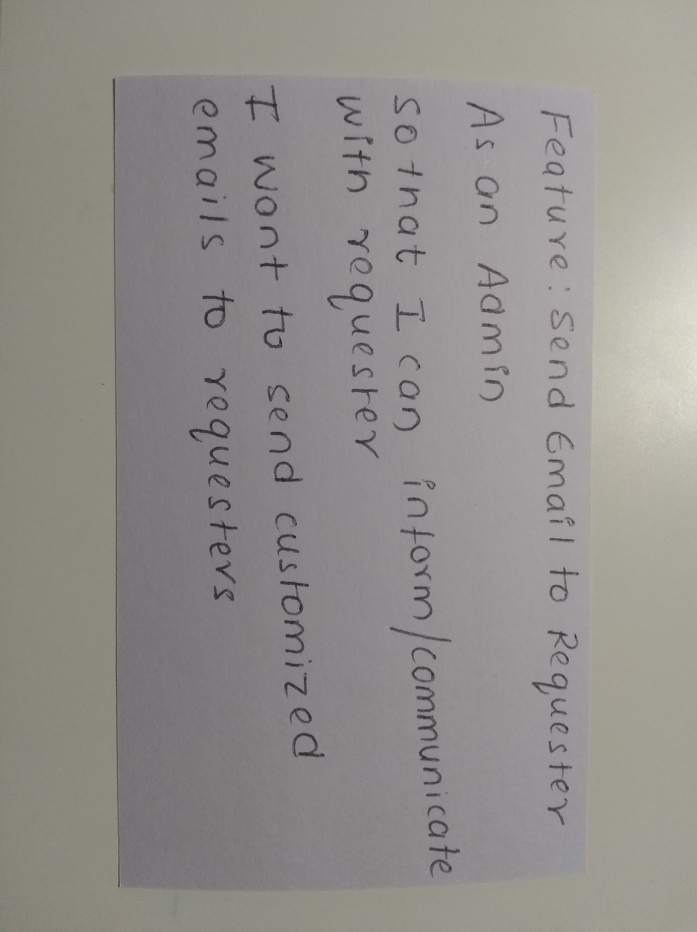
**Customer meeting:** Tuesday 10/09, 10:00 a.m at HRBB 302

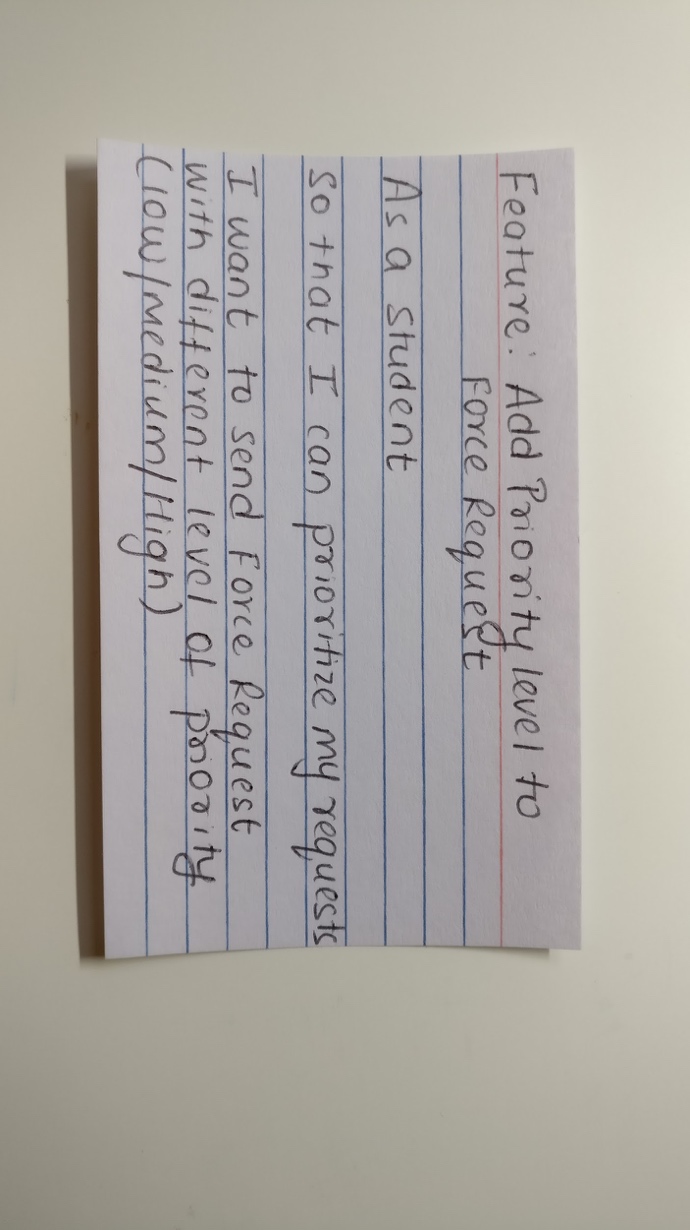
**Summary**

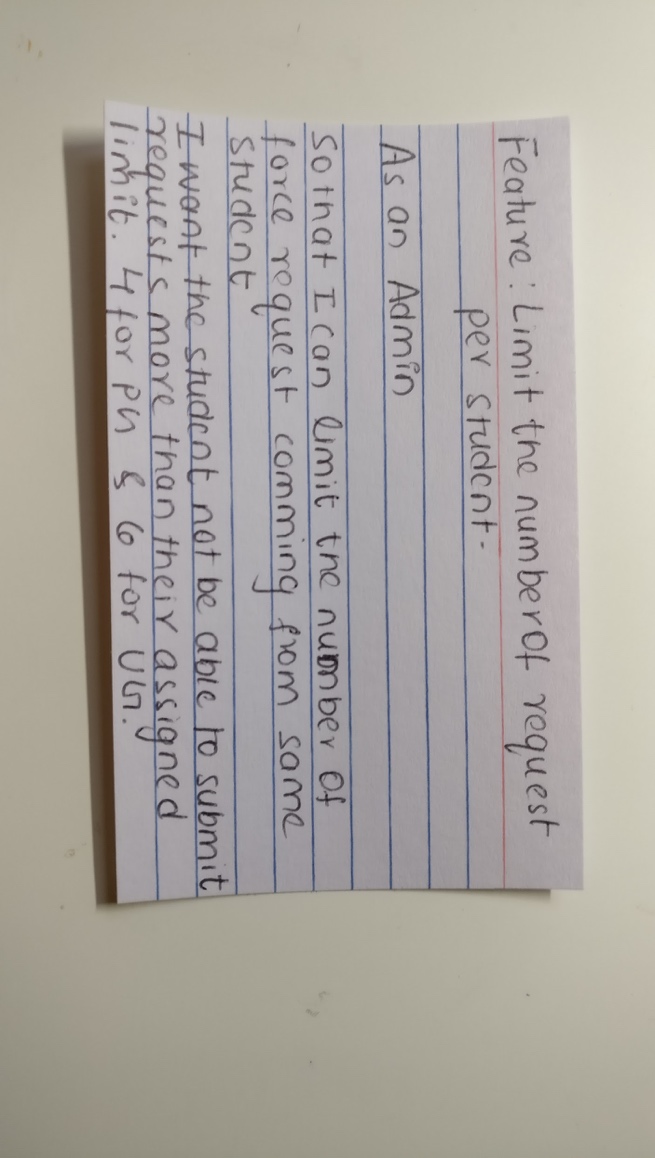
The main customer need is for the students of CSE department to submit force requests while having better control over managing the requests. The existing application developed through legacy projects meets the expectation of students submitting force requests through a three-step process. The student creates an account with the system using their email; which is then verified for their identity through directory search to ensure unique identities; account creation is confirmed via email and the user is redirected to the force request form. Future logins present the student with the view of the status of previously submitted force requests. Administrators, upon login, see the list of active force requests and are provided with the options to approve, deny or place the force requests on hold. When an admin changes the status of the force request the student receives an email notification.

Through our Iteration 0 meeting with Dr. Schaefer, we got a high-level idea of the customer’s expectations. A couple of pressing concerns addressed includes customer, having the ability to edit the status change email templates as an admin prior to sending the notification to the user. Additionally, being able to limit the number of force requests a student is allowed to submit (Graduate - 3/4 and Undergraduate 5/6 ). The customer also expressed concerns on how the system doesn’t allow user to submit force requests for the same course across different semesters (Summer + Fall). During our first interaction with the customer, we identified that most of our focus will be on the admin side providing them the capabilities to better handle the volume of force requests. The customer is also interested in having a feature that allows capture and review of the audit trail. Our stakeholders are Dr. Schaefer, Dr. Walker and the academic advisors of CSE Department. Link to the customer video

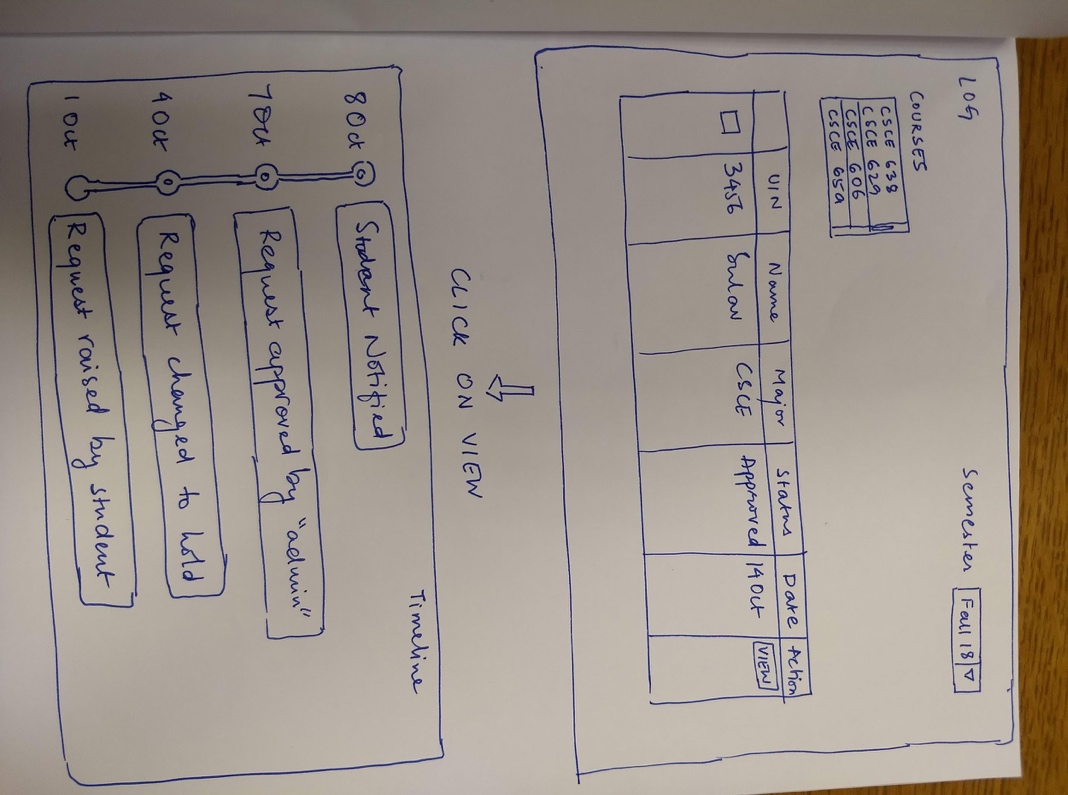
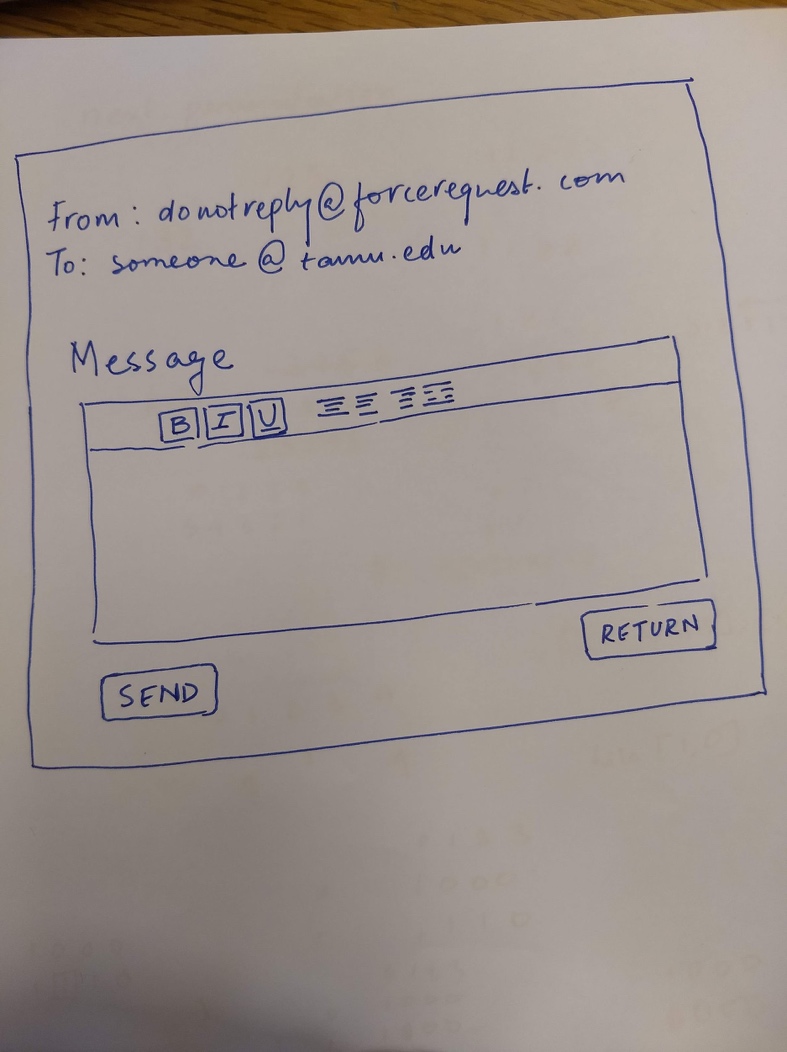
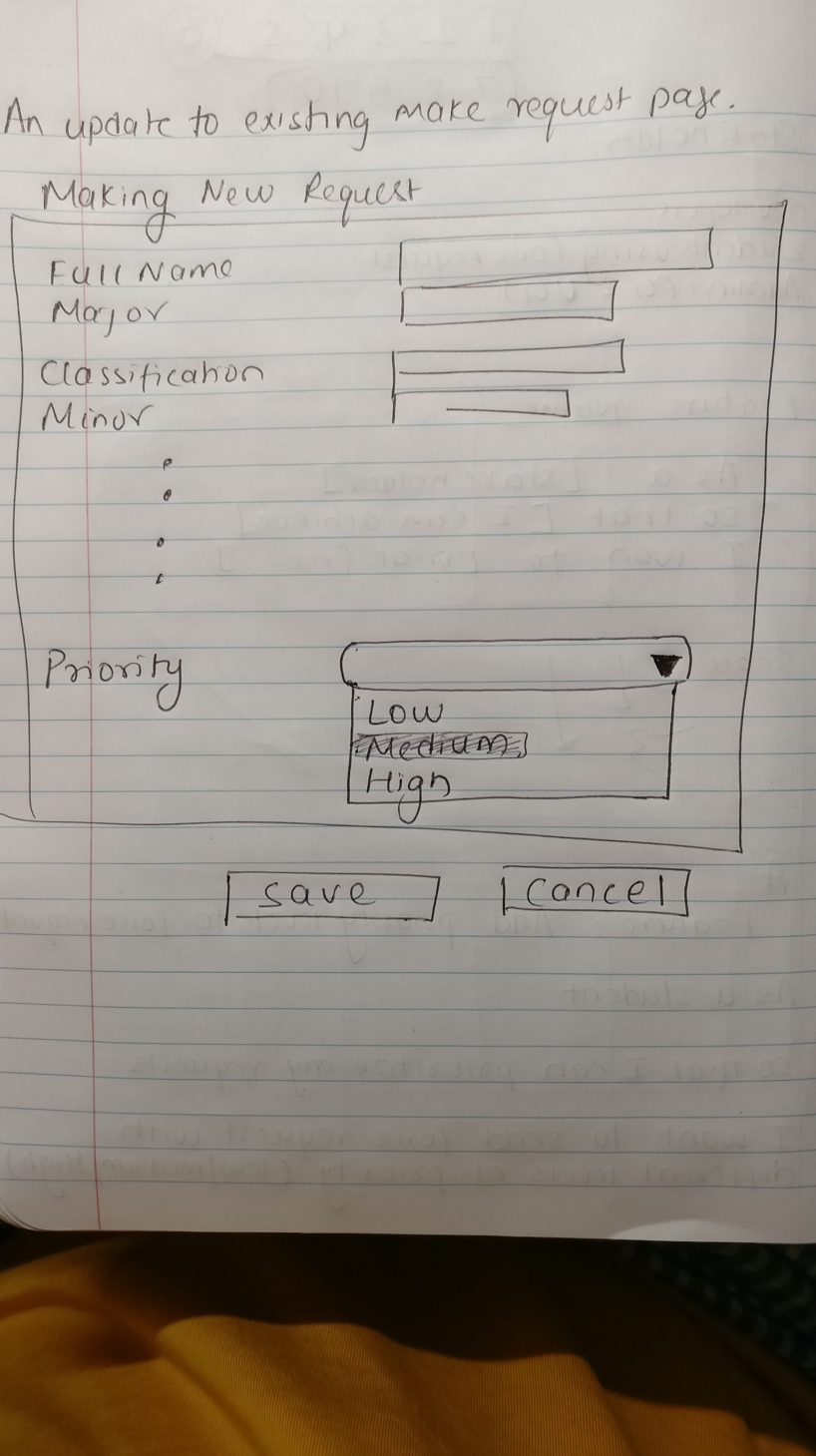
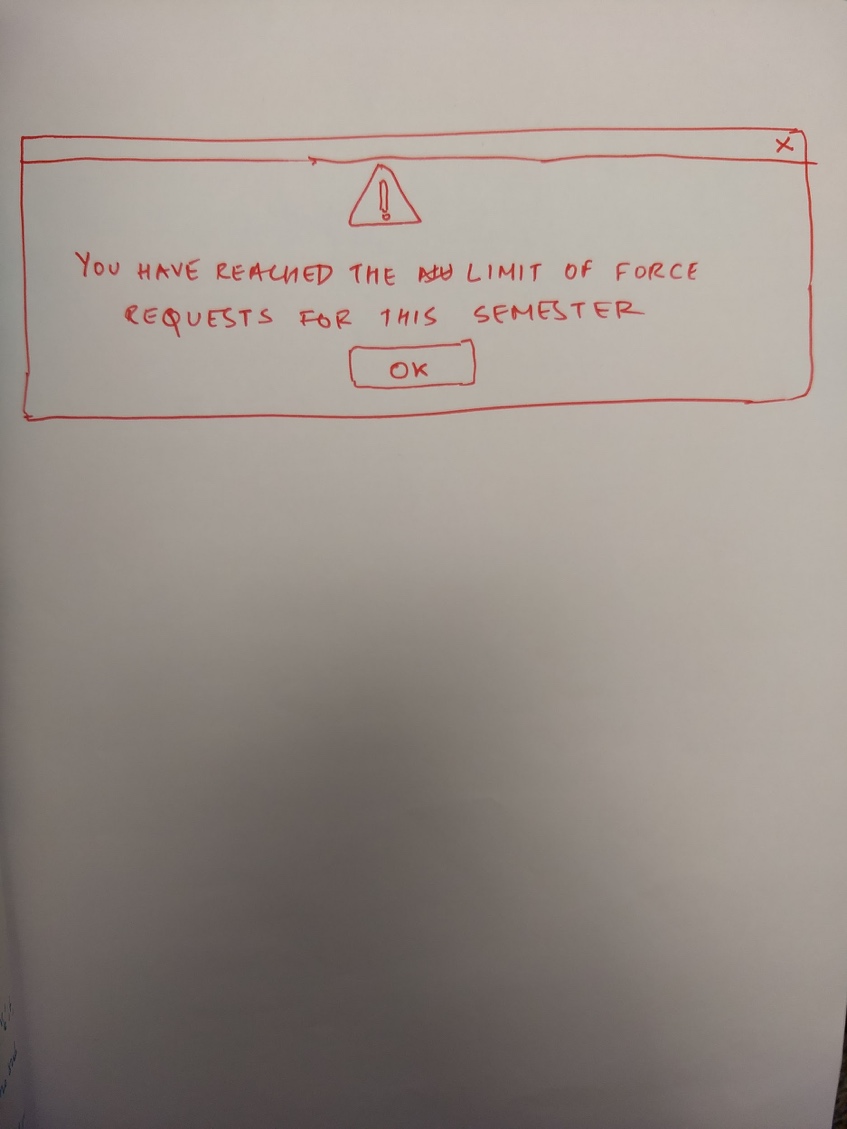
**User Stories**

1. Maintain logs
2. Customized emails



1. Prioritization of force requests  
   
2. Limit number of force requests

**User Interface**

1. Maintain logs
2. Customized emails
3. Prioritization of force requests
4. Limit number of force requests

**Pivotal Tracker:** <https://www.pivotaltracker.com/projects/2204401>

**Video:** <https://vimeo.com/294268751>

**GitHub:** <https://github.com/abkds/Force-Request-ChiUSDE>

**Legacy Code Strategy**

* In the legacy code , lot of code repetitions were encountered. We will be refactoring these code by the DRYing fashion.
* We will be analysing the flow of the code by dividing the team and using pair-programming
* In the legacy code, we found lot of validations performed at the controller level making it unnecessarily fat. We feel these validations could be moved to the client side (at javascript level) thereby creating specific and optimized controllers
* Regular unit-test method would be created for every new method integrated in the system

User Stories Improvement/Replacement

* Creating custom emails - In the previous system, an automated email was sent to the student as soon as the status is updated by the admin. However, in the new user story, admin can create custom emails for the student when a status is changed.
* Limiting the # of force requests per student - In the previous system , it was allowed for a student to raise any number of requests to the system. However, in the new user story, we will be setting restrictions on the maximum number of force requests to 4 for graduate students and 6 for undergraduate students
* Tracking the status of request : In the previous system, only the latest status of the force request was shown and in the new system we will be enhancing the feature by letting the admin track the status of force request from inception to conclusion. This will show every status of the request until the request is resolved by the admin.