

Rails test application

RAILS TEST APPLICATION

Helpdesk: A simple system for tracking incoming enquiries from customers with an easy to use interface for support staff to communicate with clients.

INTRODUCTION

Please ensure your application uses Rails4 with HAML or SLIM as your markup language. The system should be DB agnostic. You're free to use NoSql when and where it may be needed.

Build the application in the manner *you feel* most appropriate to accomplish all the tasks outlined in the specification. We'd appreciate your BDDness, DRYness and KISSness.

When the project is complete, send us github link to your project.

CUSTOMER INTERFACE

- Customers do not need to login to submit a query - they simply will need to provide their name and email address, the subject of their enquiry and the full body of the request. That page should have root url ("/").
- Each new ticket is assigned a unique reference in the format similar to Ticket-ABC-123-ABC-123-ABC (where: ABC is random 3-char string, 123-random 3-digit number).
- When a new ticket is added, the customer should receive an email confirming their request has been received along with their unique reference. This email should also include a URL which can be used to view the ticket's history and update tickets.
- *Additional points:*
 - A. user may want to submit few screenshots - and they will be visible for staff as clickable thumbnails;
 - B. protect form from spam without captcha;
 - C. user may select the the issue relates department

STAFF INTERFACE

- Each member of staff team should have a username & password, used to login to the system. There should be 2 different roles:
 - admin, which can add new member and edit/delete other member posts
 - member, can response to customer post, edit own response
- Staff can view a list of all unassigned, open tickets and take ownership.
- Staff should be able to reply to the ticket by simply entering their response into a text field. As part of the same action, they should also be able to change the owner or status.
- All replies should be e-mailed to the client.
- Each ticket should be assigned a status: 'Waiting for Staff Response', 'Waiting for Customer', 'On Hold', 'Cancelled' or 'Completed' - status(-es) may wish to be added in the future.

- When a ticket is first created or updated by the customer, it should be assigned the 'Waiting for Staff Response' status.
- The interface should have four main views:
 1. New unassigned tickets
 2. Open Tickets
 3. On hold tickets
 4. Closed Tickets
 5. Manage members (visible only for admin)
 6. Change password page
- Staff should be able to quickly open up a new ticket by entering it's reference number into a search field.
- *Additional points:*
 - D. Staff may want to add few screenshots too and they will be visible as clickable thumbnails;
 - E. They may also wish to search based on the ticket subject.
 - F. Also please allow staff members to look for words or combinations of words found in the ticket.
 - G. All changes to the status, owner and response should be tracked and visible to all staff members.
 - H. Also, add possibility to admin to add/edit/delete departments, change department of staff members; add possibility to separate departments issues for staff members, so members of one department can view only own department issues and add possibility for staff members to change department.
 - I. Also, you can use SocketIO library for realtime update users view.