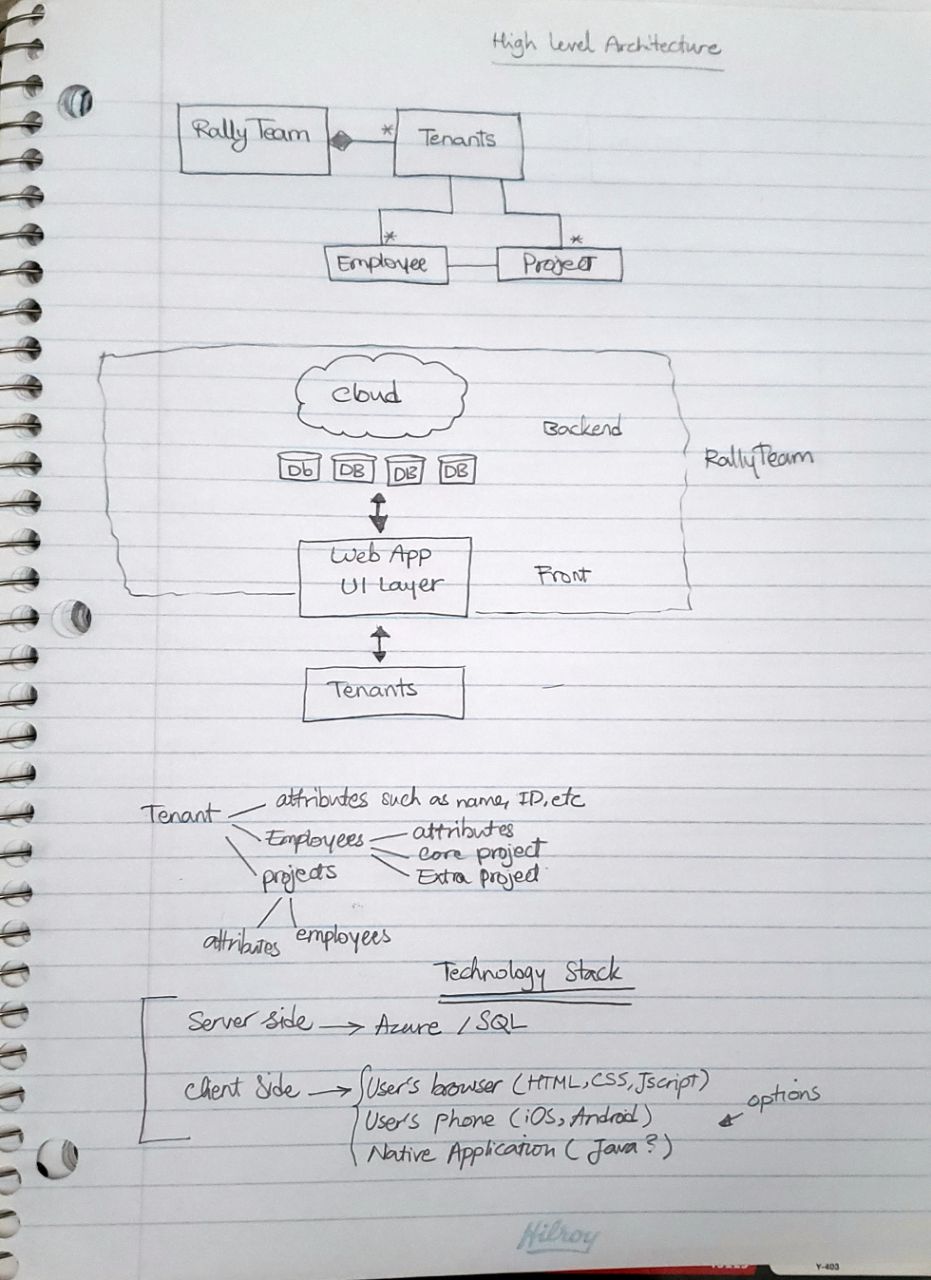
Rallyteam Technical Exercise

Design

Based on your understanding of Rallyteam, how would you design the system from front to back? Please provide an architecture diagram and explain what technologies you would use and why? Feel free to register and log into our dev environment ([https://dev.rallyteam.com](https://dev.rallyteam.com/) ) to familiarize yourself with Rallyteam. *You will receive an invite to join the application*.

Constraint: Each company (aka tenant) requires that their data be stored in a separate database.

**A high-level architectural view of the company would be something similar to the diagram below:**



**The technology stack consists of server and client. On the server side Rallyteam can make use of cloud computing to efficiently manage the client side. The justification is:**

* **Elastic computing option (The number of clients, their separate Db’s, projects and data can be handled in a costly efficient manner.**
* **24/7 service availability/support/backup option**
* **Less Physical database and storage hassle**

**As Microsoft already has provided an economic option, Azure seems to be a perfect solution. As for the database solution MySQL works fine, It is fully integrated with Azure and is standard.**

**On the client side, users will use their browser (chrome, ie, Safari,etc) or their phone browsers. The web application should support both PC, Mac and respectively iOs and Android. A variety of programming languages and scripting can be used in this case.**

Analysis

Rallyteam’s core concept is the Opportunity Market. This is where users login to look for potential opportunities to work on and/or post opportunities asking for help. For some of our pilots, user engagement is starting to drop - users log in, browse some opportunities and don’t come back.

How would you address this problem?

**The idea is to make the market:**

* **More attractive and entertaining (UI, adding widgets, etc)**
* **More challenging and competitive**
* **More fruitful ( pointing system, prize and reward)**

**Please check below for some suggestions.**

What are some solutions you would propose to encourage users to post and join opportunities?

* **Incentives to encourage the users to sign up for jobs ( such as prizes, cash,etc)**
* **Pointing system/prestige level. Implementing a pointing system that promotes users to the next level that eventually announces a winner or prestige prize of the month can be a good idea.**
* **Making the job/project more competitive by setting timer on the job, and let users “bid” for the time that they can accomplish it.**

Technical Challenges

For submission, please be ready to present your work and publish your source code in Github

1. Refactoring Exercise - There will be times when features have been developed extremely fast and need to refactored. Please download the following repository, complete the To Do’s and refactor as you see necessary.   
     
   Shapes Calculator: <https://github.com/Rallyteam/AreaCalculator>

*Note: Feel free to re-write/refactor in the language of your choice as well.*

1. API Integration Exercise – In Rallyteam, we're very proud that our system can be easily integrated with other systems. We've created a solid API documentation that we give to our customers their data and present it the way they want.  
     
   For this exercise, you're working for an organization that wants to tap in to Rallyteam's jobs API. You're responsible for building a web application that lists out all jobs in Rallyteam that are on-site and list out all the jobs' hiring manager.  
     
   *Use the account that you created when you were invited to the application for authentication to the API.*  
     
   API Documentation: <https://dev.rallyteam.com/swagger/ui/index>*The focus of this exercise is on the data and not on the User Interface so do not spend too much time building a polished UI.*  
     
   Requirements:
   * Build a web-application that lists out all the available jobs
   * When you click on a job, you should be able to display the hiring manager's display name and email address