CMPE 275
Spring 2019
Enterprise Application Development

Team 14 - OpenHack
Project Report

SAN JOSÉ STATE
UNIVERSITY

Submitted to –
Prof. Charles Zhang

on
05/21/2019

Apurav Gupta	013728871	apurav.gupta@sjsu.edu
Mayur Barge	013722488	mayur.barge@sjsu.edu
Sheena Gupta	013591981	sheena.gupta@sjsu.edu
Varun Jain	013719108	varun.jain@sjsu.edu

Table of Contents

Topic	Page No.
Introduction	3
Motivation	3
Design	6
Technology Stack	7
Features	8
Test Plan and Results	9
Lessons Learned	10
Future Work	13

Chapter 1: Introduction

OpenHack is cloud hosted service to create and organize hackathon events. This service can be accessed from browser as web application. There are following two personas for OpenHack service -

1. Administrator
   With current implementation any user with San Jose State University email address is considered as administrator of application. Administrators have highest level of permissions on web application and can create Hackathons. Entire lifecycle of hackathon like opening/closing hackathon for code submission and finalizing hackathon is managed by administrator. Administrator can get additional details about the hackathon like earning report.
2. Hacker
   Any user with valid email address other than San Jose State University email address is considered as hacker on OpenHack. A hacker can create organization so that hackathon can be sponsored by organization. The owner of organization can approve/reject membership requests from other hackers of OpenHack. Hacker gets discount on hackathon fee if organization that he/she is member of is sponsoring respective hackathon. Any hacker can participate in hackathon by enrolling as team. Once payment for all team members in a team is done the team is finalized and can submit code as URL.

The OpenHack application client is developed with React and server-side implementation is in Java.

Chapter 2: Motivation

OpenHack service is developed with following goals in mind

1. Distributed component design with MVC architecture style
2. Scalability
3. Modern and emerging application development technologies

OpenHack application allowed us to learn principals, patterns and methodologies that we have learned in the class which includes

1. Dependency Injection
2. Aspect Oriented Programming
3. Model, View, Controller
4. Object Relational Mapper
5. Transactions
6. Security

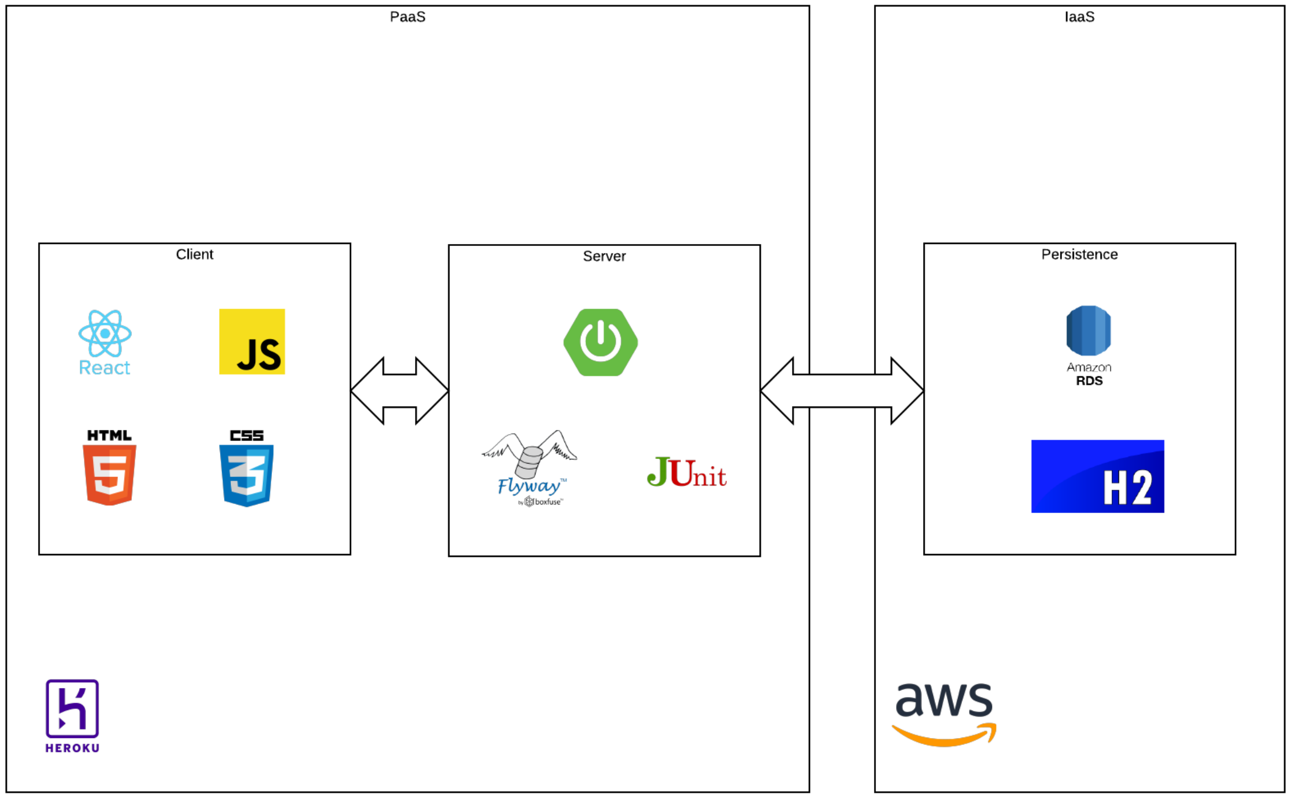
Chapter 3: Design

Chapter 3.1: High Level Design

OpenHack application is divided into multiple component as shown in below diagram. Both client and server application are managed in separate version controller repository for agile development. This helped us to develop features independently. React JavaScript library is used to build user interfaces for client-side application. React is suitable for fetching rapidly changing data. Server-side application is developed in Spring boot with MVC architecture. REST based services are divided into multiple components and layers. For the management of different versions of database schema, we have used Flyway database migration tool. Junit is used as testing framework.

MySQL database is used as persistence mechanism for OpenHack application. MySQL is hosted onto cloud hosted service RDS provided by AWS. H2 database is used for unit and integration test cases as in-memory database.

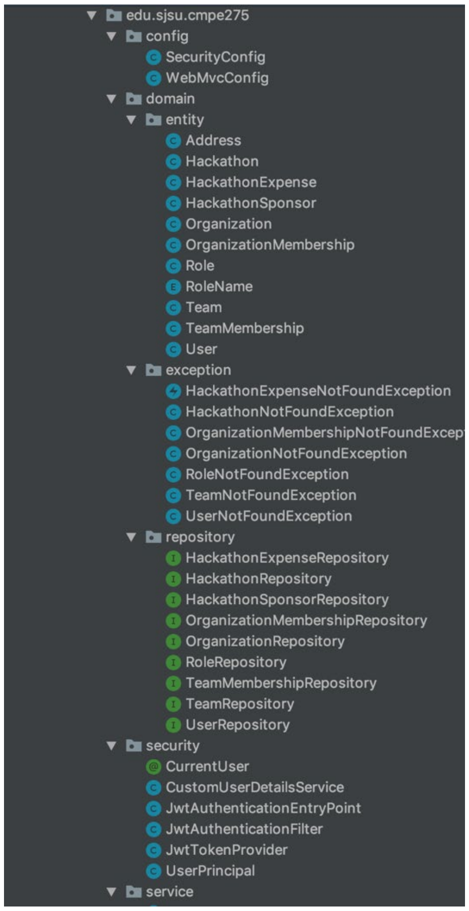
For storing profile picture, we have used AWS S3 service. AWS S3 and email are two additional services used in application as helper services.



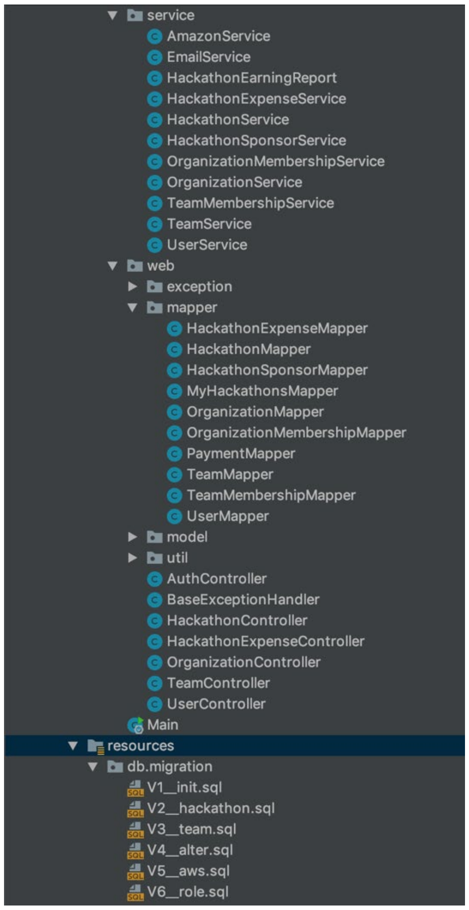
Chapter 3.1: Component Level Design

Server-side application is divided into following three layers

1. Domain
   This layer consist anything related to persistence layer
   a. Entity
   b. Persistence Exceptions
   c. JPA
2. Service
   This consist of business layer which will interact with JPA
   a. Security
   b. Amazon Service
   c. Email Service
   d. Other business layer services
3. Controller
   This consist of REST controllers to handle requests. Additionally, this layer has data transfer objects and related mapper
   a. Web Exceptions
   b. Model and Mapper
   c. Controller



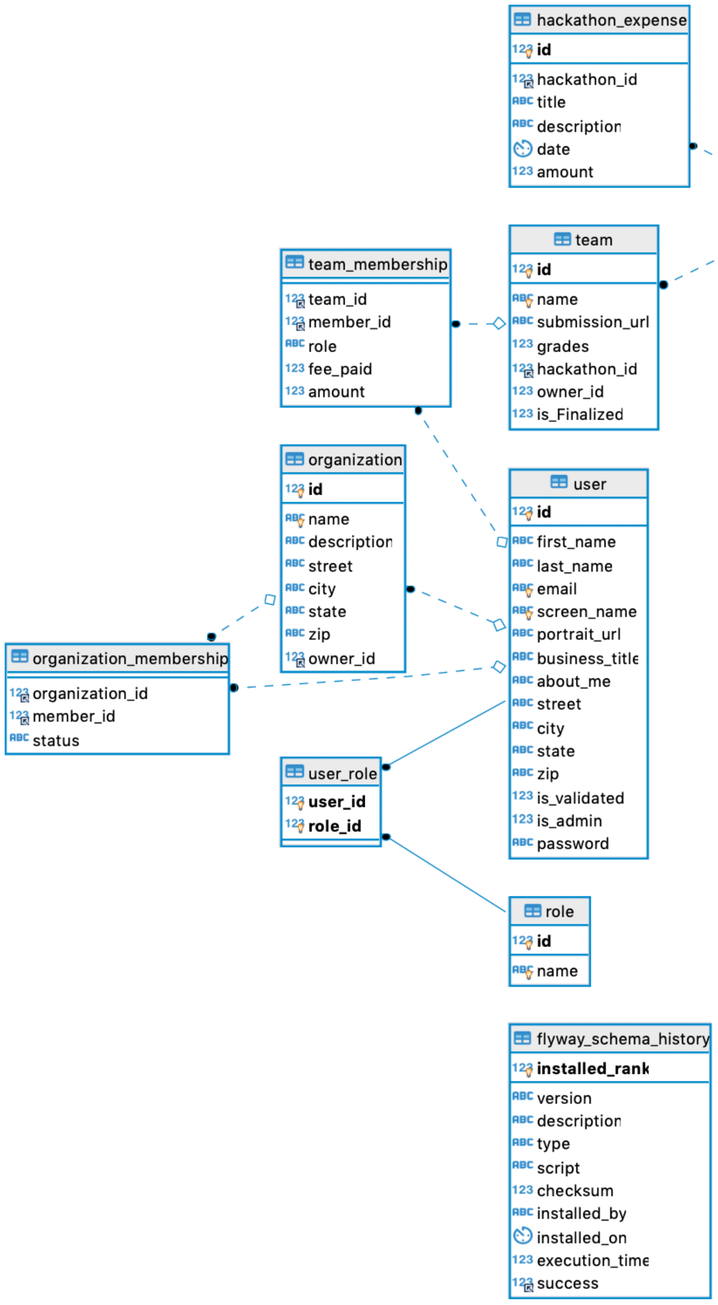
```
▼ edu.sjsu.cmpe275
  ▼ config
      SecurityConfig
      WebMvcConfig
  ▼ domain
    ▼ entity
        Address
        Hackathon
        HackathonExpense
        HackathonSponsor
        Organization
        OrganizationMembership
        Role
        RoleName
        Team
        TeamMembership
        User
    ▼ exception
        HackathonExpenseNotFoundException
        HackathonNotFoundException
        OrganizationMembershipNotFoundExcep
        OrganizationNotFoundException
        RoleNotFoundException
        TeamNotFoundException
        UserNotFoundException
    ▼ repository
        HackathonExpenseRepository
        HackathonRepository
        HackathonSponsorRepository
        OrganizationMembershipRepository
        OrganizationRepository
        RoleRepository
        TeamMembershipRepository
        TeamRepository
        UserRepository
  ▼ security
      CurrentUser
      CustomUserDetailsService
      JwtAuthenticationEntryPoint
      JwtAuthenticationFilter
      JwtTokenProvider
      UserPrincipal
  ▼ service
```



```
    ▼ service
        AmazonService
        EmailService
        HackathonEarningReport
        HackathonExpenseService
        HackathonService
        HackathonSponsorService
        OrganizationMembershipService
        OrganizationService
        TeamMembershipService
        TeamService
        UserService
    ▼ web
      ▶ exception
      ▼ mapper
          HackathonExpenseMapper
          HackathonMapper
          HackathonSponsorMapper
          MyHackathonsMapper
          OrganizationMapper
          OrganizationMembershipMapper
          PaymentMapper
          TeamMapper
          TeamMembershipMapper
          UserMapper
      ▶ model
      ▶ util
        AuthController
        BaseExceptionHandler
        HackathonController
        HackathonExpenseController
        OrganizationController
        TeamController
        UserController
      Main
  ▼ resources
    ▼ db.migration
        V1__init.sql
        V2__hackathon.sql
        V3__team.sql
        V4__alter.sql
        V5__aws.sql
        V6__role.sql
```

Chapter 3.2: Database schema

Following is the database schema that we have used for application. Hackathon, User, Organization and Team are main database schema table. Other tables are used for relationship mapping.



Chapter 3.3: REST API

Following are the API endpoints that we have developed for server-side application. /users, /organizations and /hackathons are considered as top-level resource while /memberships, /teams are considered as sub resources for /organizations and /hackathon respectively.

User		
POST	/users	Create user
GET	/users	Get all users
PATCH	/users/{id}	Update user profile
GET	/users/{id}	Get user profile
GET	/users/{id}/hackathons	Get hackathons for user
GET	/users/{id}/memberships	Get memberships of particular user

Organization		
POST	/orgs	Create organization
GET	/orgs	Get all organizations (Searchable organizations)
GET	/orgs/{id}	Get organization details
GET	/orgs/{id}/members	Get members of organization
POST	/orgs/{id}/memberships	Request New Membership
GET	/orgs/{id}/memberships	Get all memberships
PUT	/orgs/{id}/memberships	Change membership

Hackathon		
POST	/hackathons	Create hackathon
GET	/hackathons	Get all hackathons (Searchable hackathons)
GET	/hackathons/{id}	Get particular hackathon details
PATCH	/hackathons/{id}	Hackathon admin operations (open/close/finalize)
POST	/hackathons/{id}/teams	Register team for hackathon
GET	/hackathons/{id}/teams	Get all teams for hackathon
GET	/hackathons/{id}/teams{id}	Get particular team details
PATCH	/hackathons/{id}/teams/{id}	Code submission / Grading /is finalized for team
GET	/hackathons/{id}/teams/{id}/payment	Get Payment amount
POST	/hackathons/{id}/teams/{id}/payment	Process Payment

Chapter 4: Technology Stack

Frontend

1. ReactJS (JavaScript)
2. HTML5
3. CSS3
4. npm

Server

1. Spring Boot (Java 8)
2. Junit
3. Flyway
4. Lombok

Persistence

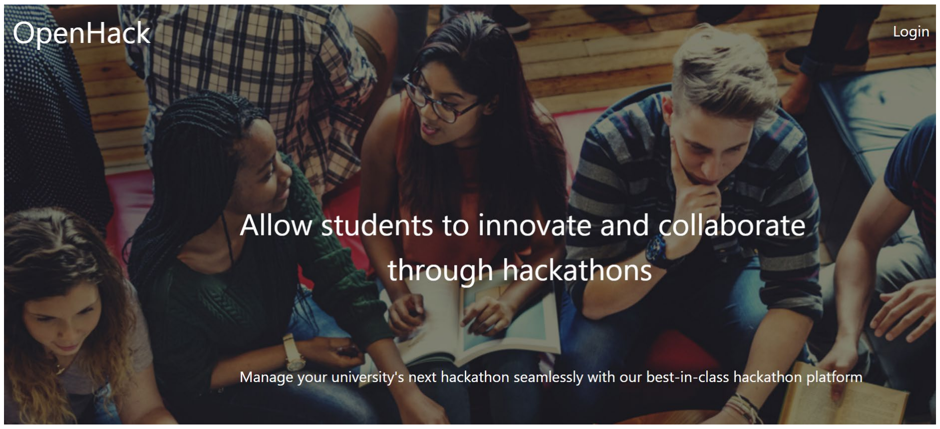
1. MySQL 8
2. H2
3. AWS S3

Deployment

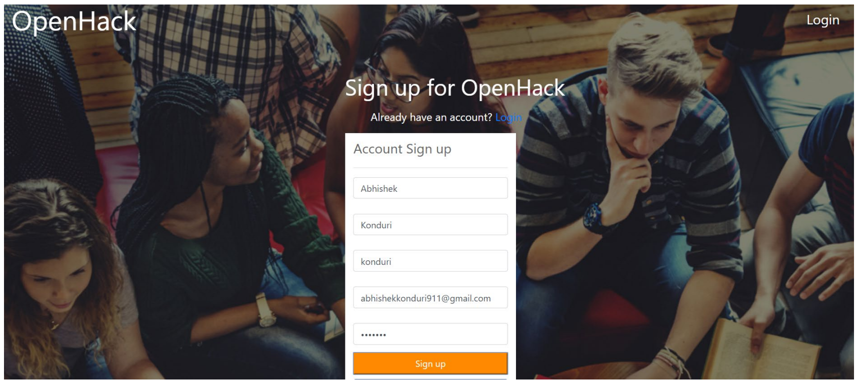
1. Heroku
2. AWS RDS

Chapter 5: Features

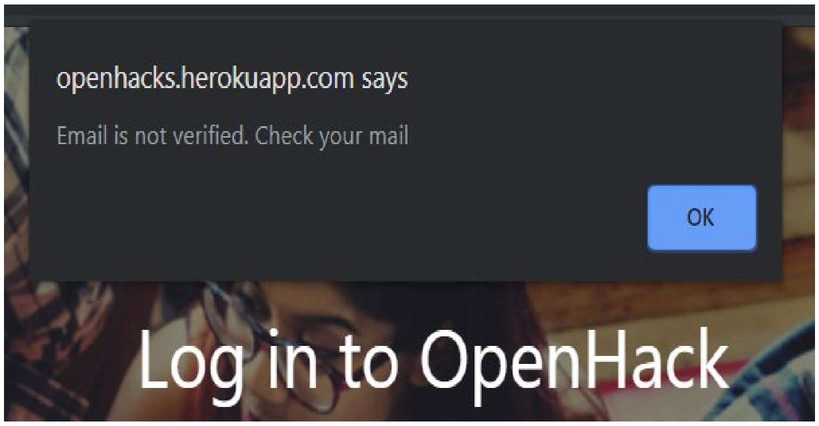
Landing Page


OpenHack
Login
Allow students to innovate and collaborate through hackathons
Manage your university's next hackathon seamlessly with our best-in-class hackathon platform

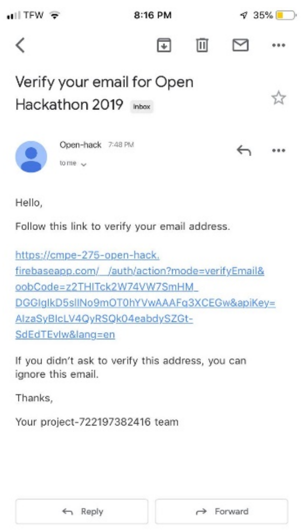
Login Page



Validation when trying to login without email verification (Refresh the page after verification done)


openhacks.herokuapp.com says

Email is not verified. Check your mail



Verify your email for Open Hackathon 2019 Inbox

Open-hack 7:48 PM
to me

Hello,

Follow this link to verify your email address.

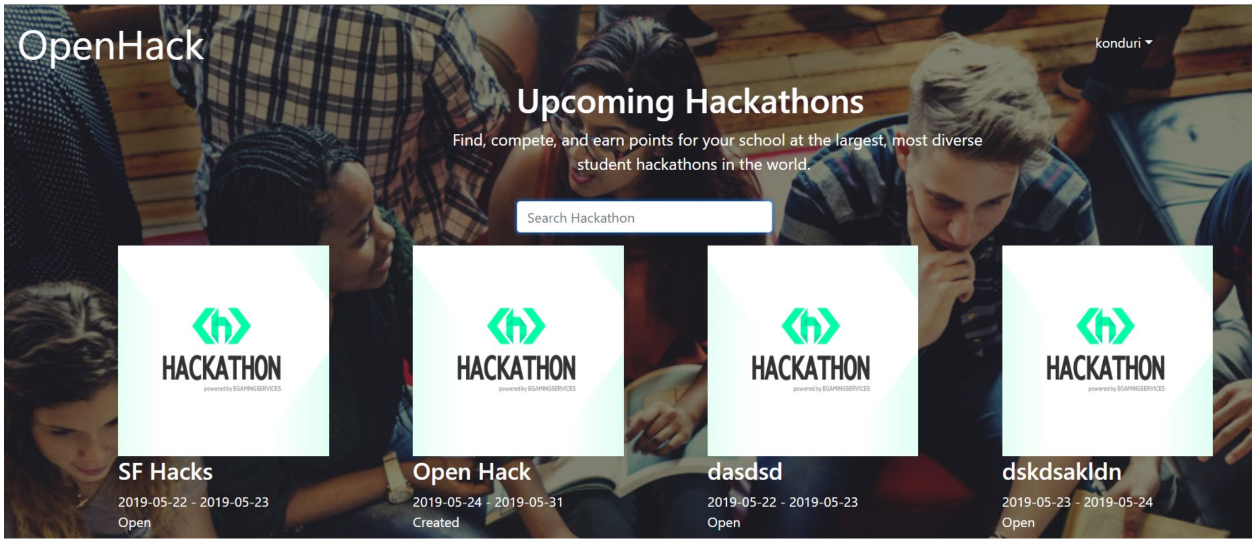
https://cmpe-275-open-hack.firebaseapp.com/__/auth/action?mode=verifyEmail&oobCode=z2THITck2W74VW7SmHM_DGGIglkD5slINo9mOT0hYVwAAAFq3XCEGw&apiKey=AIzaSyBIcLV4QyRSQk04eabdySZGt-SdEdTEvlw&lang=en

If you didn't ask to verify this address, you can ignore this email.

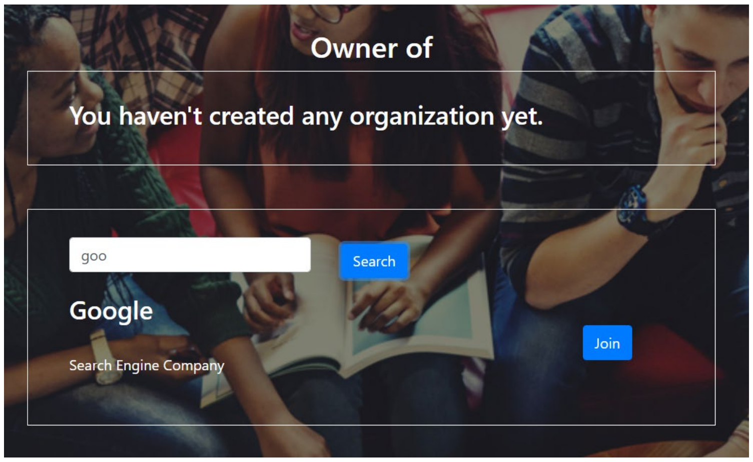
Thanks,

Your project-722197382416 team

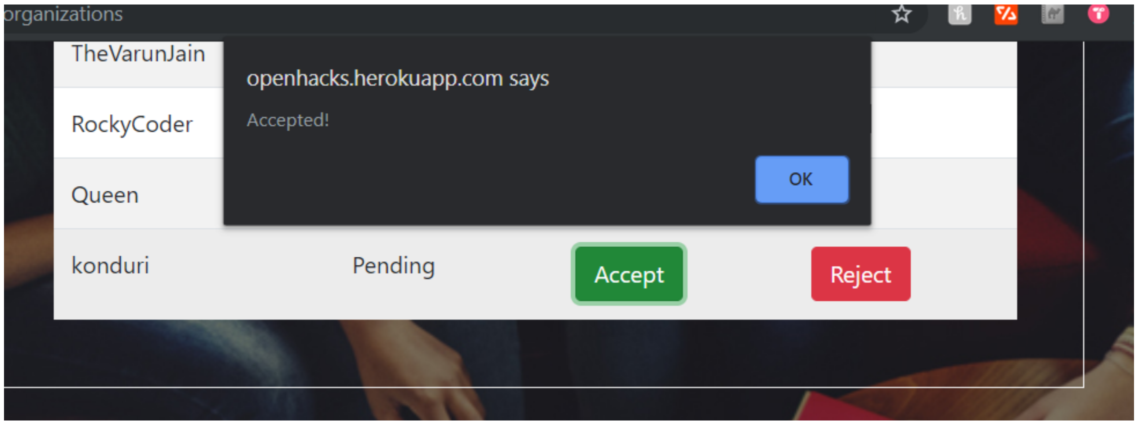
As we log in, we get home page with all public hackathons available



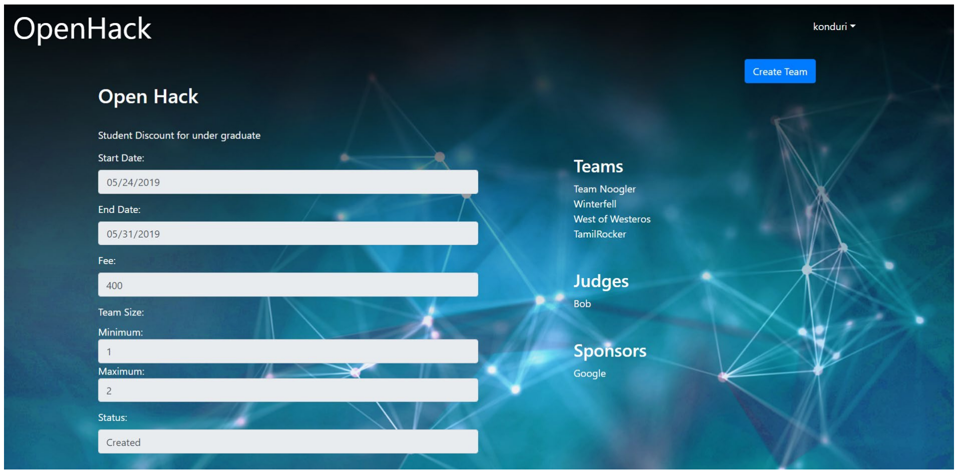
Organization Join Invitation



Organization Invitation Accept



A non admin user can create a team by opening any hackathon.



Open Hack

Student Discount for under graduate

Start Date: 05/24/2019

End Date: 05/31/2019

Fee: 400

Team Size:

Minimum: 1

Maximum: 2

Status: Created

Teams

Team Noogler
Winterfell
West of Westeros
TamilRocker

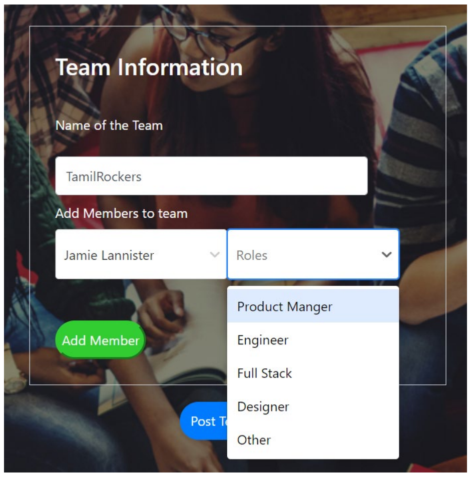
Judges

Bob

Sponsors

Google

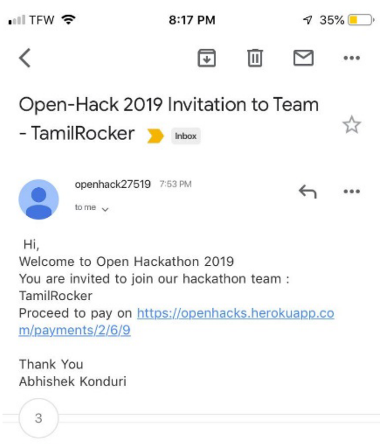
Creating a team will send an email to all the team members.



Name of the Team: TamilRockers

Add Members to team: Jamie Lannister | Roles

Product Manger
Engineer
Full Stack
Designer
Other



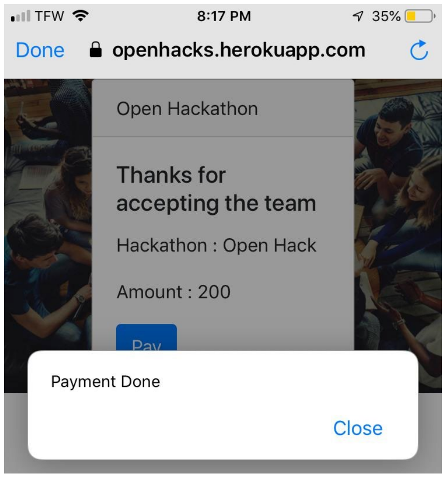
Open-Hack 2019 Invitation to Team - TamilRocker

openhack27519 7:53 PM
to me

Hi,
Welcome to Open Hackathon 2019
You are invited to join our hackathon team : TamilRocker
Proceed to pay on https://openhacks.herokuapp.com/payments/2/6/9

Thank You
Abhishek Konduri

The payment link in email will re direct to payment page. Once Payment done, Team is finalized.



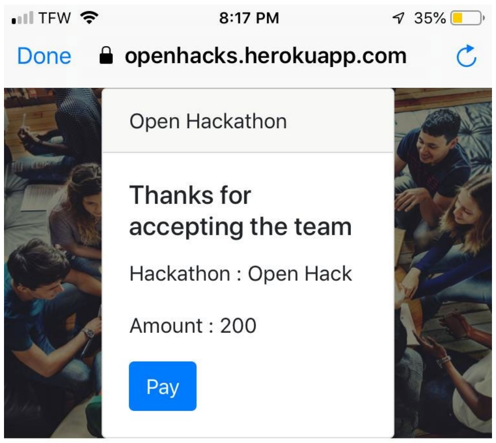
Open Hackathon

Thanks for accepting the team

Hackathon : Open Hack

Amount : 200

Payment Done



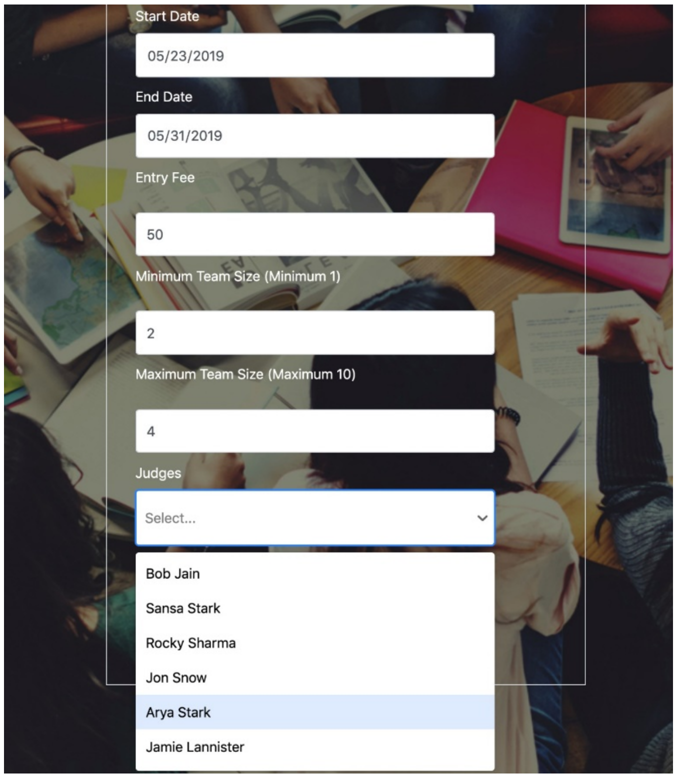
Open Hackathon

Thanks for accepting the team

Hackathon : Open Hack

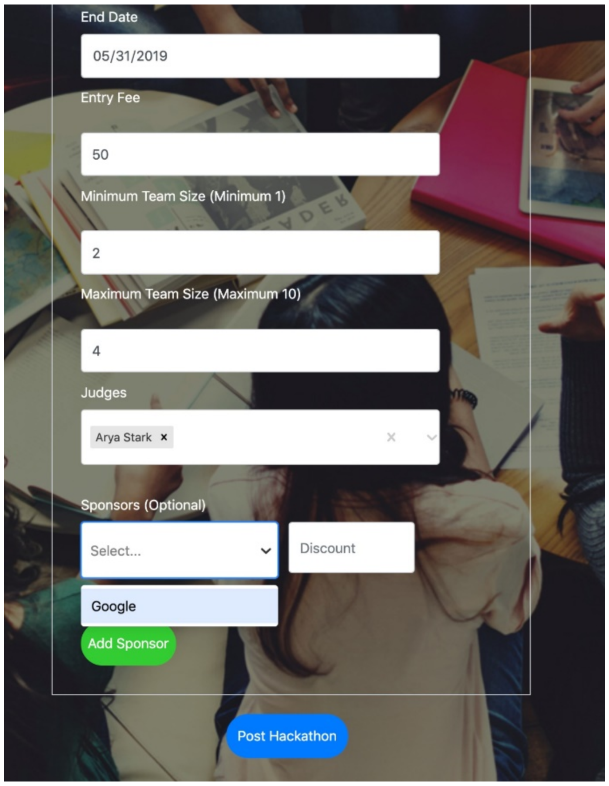
Amount : 200

Create Hackathon
Only an admin user can create hackathon



Start Date
05/23/2019
End Date
05/31/2019
Entry Fee
50
Minimum Team Size (Minimum 1)
2
Maximum Team Size (Maximum 10)
4
Judges
Select...

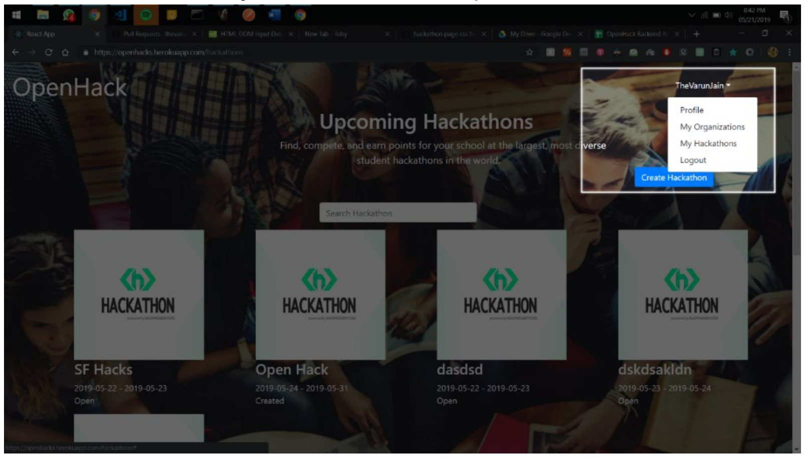
Bob Jain
Sansa Stark
Rocky Sharma
Jon Snow
Arya Stark
Jamie Lannister



End Date
05/31/2019
Entry Fee
50
Minimum Team Size (Minimum 1)
2
Maximum Team Size (Maximum 10)
4
Judges
Arya Stark ×
Sponsors (Optional)
Select... Discount
Google

Admin Page for managing hackathons

This can be find in “My hackathon” from drop down



Upcoming Hackathons

Find, compete, and earn points for your school at the largest, most diverse student hackathons in the world.

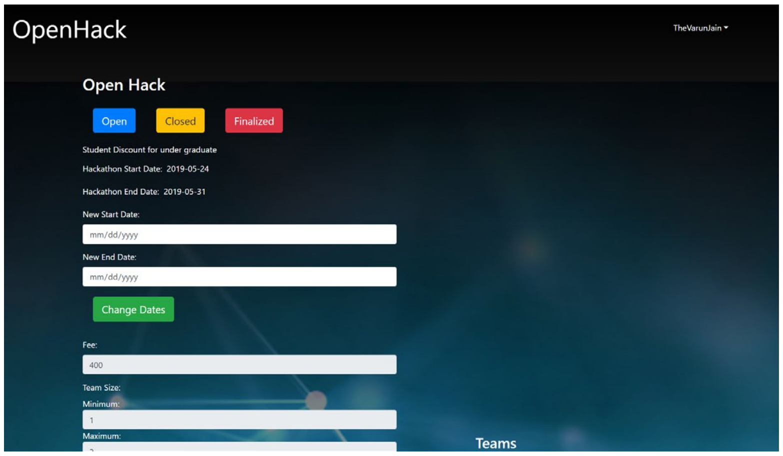
SF Hacks
2019-05-22 - 2019-05-23
Open

Open Hack
2019-05-24 - 2019-05-31
Created

dasdsd
2019-05-22 - 2019-05-23
Open

dskdsakldn
2019-05-23 - 2019-05-24
Open

Hackathon Page with changing states and dates
Using which hackathon state can be changed



Open Hack

Student Discount for under graduate

Hackathon Start Date: 2019-05-24

Hackathon End Date: 2019-05-31

New Start Date: mm/dd/yyyy

New End Date: mm/dd/yyyy

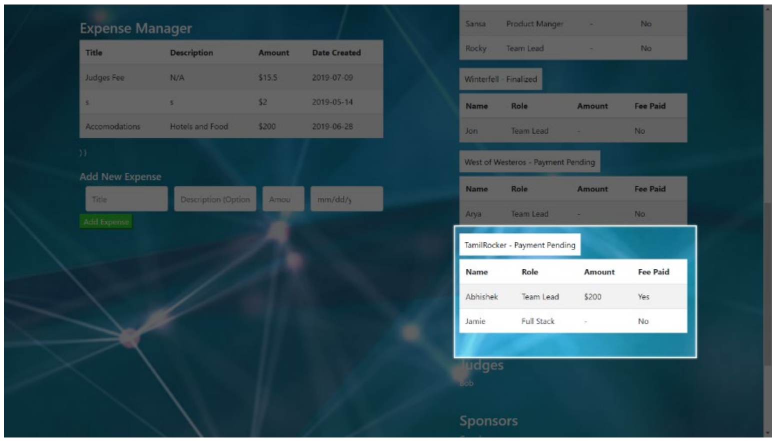
Fee: 400

Team Size:

Minimum: 1

Maximum:

In the same page we can find, all the teams participated with their team, members and status of fee
REGISTRATION FEE ANF PAYMENT REPORT



Expense Manager

Title	Description	Amount	Date Created
Judges Fee	N/A	$15.5	2019-07-09
s	s	$2	2019-05-14
Accomodations	Hotels and Food	$200	2019-06-28

Sansa	Product Manger	-	No
Rocky	Team Lead	-	No

Winterfell - Finalized

Name	Role	Amount	Fee Paid
Jon	Team Lead	-	No

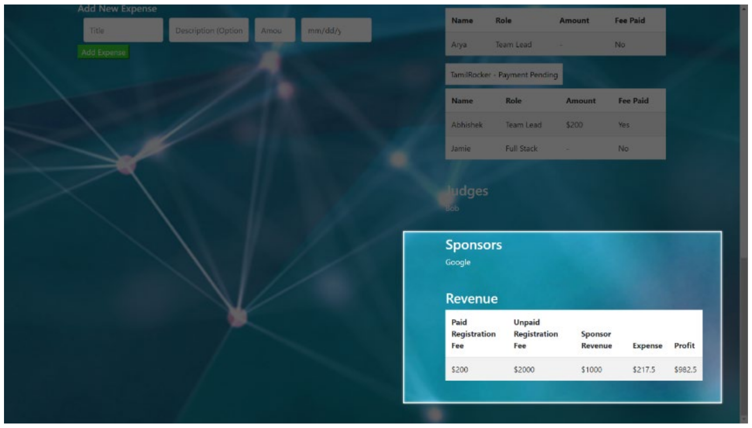
West of Westeros - Payment Pending

Name	Role	Amount	Fee Paid
Arya	Team Lead	-	No

TamilRocker - Payment Pending

Name	Role	Amount	Fee Paid
Abhishek	Team Lead	$200	Yes
Jamie	Full Stack	-	No

Earning Report



Name	Role	Amount	Fee Paid
Arya	Team Lead	-	No

TamilRocker - Payment Pending

Name	Role	Amount	Fee Paid
Abhishek	Team Lead	$200	Yes
Jamie	Full Stack	-	No

Judges

Bob

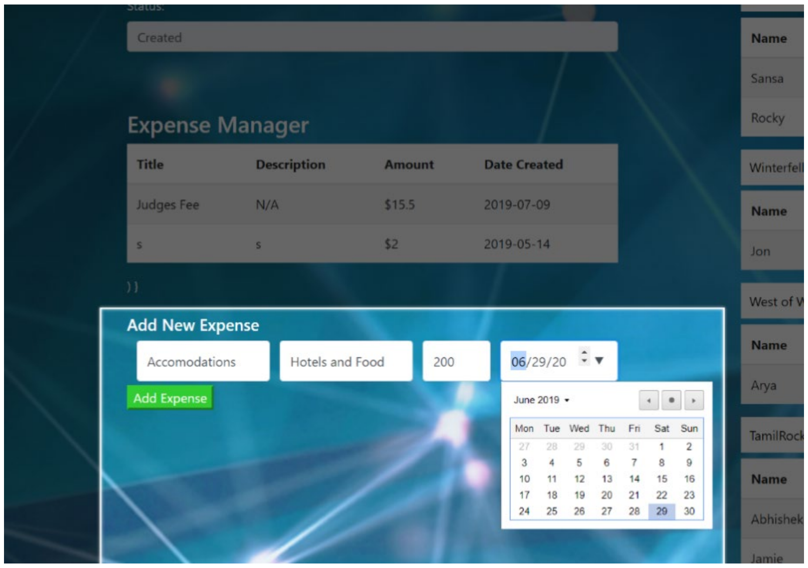
Sponsors

Google

Revenue

Paid Registration Fee	Unpaid Registration Fee	Sponsor Revenue	Expense	Profit
$200	$2000	$1000	$217.5	$982.5

Bonus Feature – Per Hackathon Expenses Report



Expense Manager

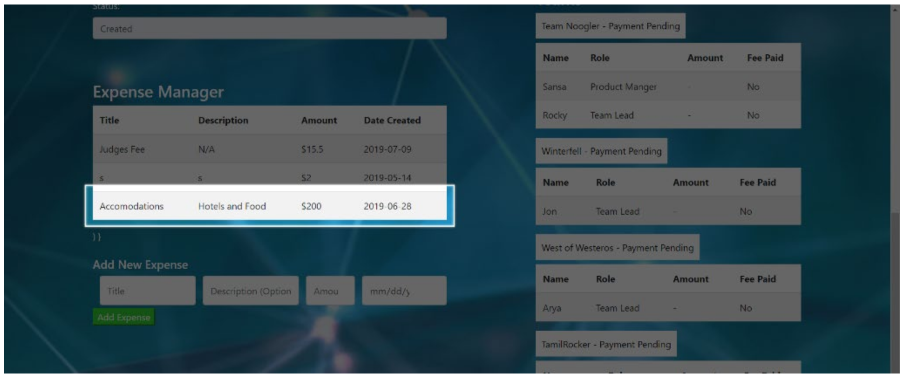
Title	Description	Amount	Date Created
Judges Fee	N/A	$15.5	2019-07-09
s	s	$2	2019-05-14

} }

Add New Expense

Accomodations | Hotels and Food | 200 | 06/29/20

Added in the repor



Expense Manager

Title	Description	Amount	Date Created
Judges Fee	N/A	$15.5	2019-07-09
s	s	$2	2019-05-14
Accomodations	Hotels and Food	$200	2019-06-28

} }

Add New Expense

Team Noogler - Payment Pending

Name	Role	Amount	Fee Paid
Sansa	Product Manger	-	No
Rocky	Team Lead	-	No

Winterfell - Payment Pending

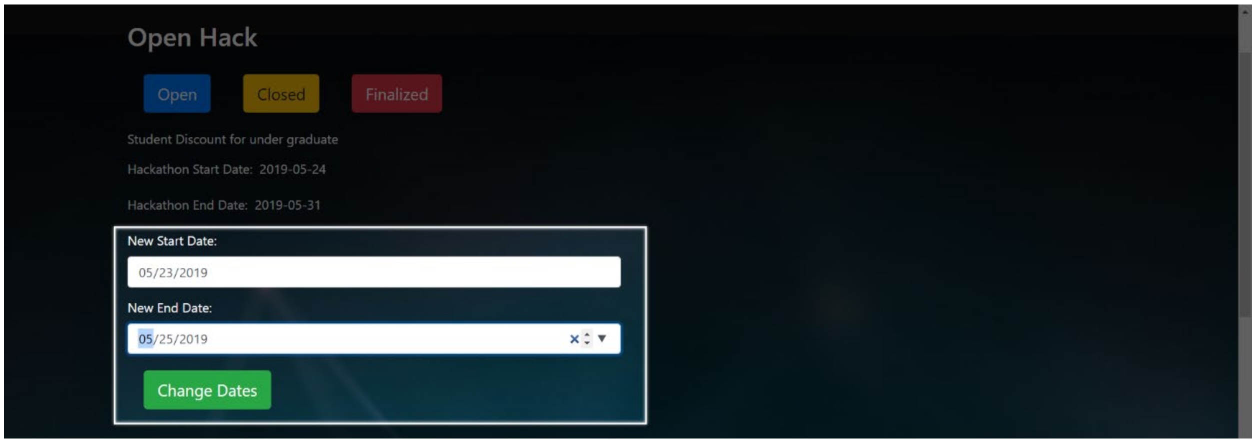
Name	Role	Amount	Fee Paid
Jon	Team Lead	-	No

West of Westeros - Payment Pending

Name	Role	Amount	Fee Paid
Arya	Team Lead	-	No

TamilRocker - Payment Pending

Changing Hackathon Dates (Reload Page to see updated result)



Student Discount for under graduate

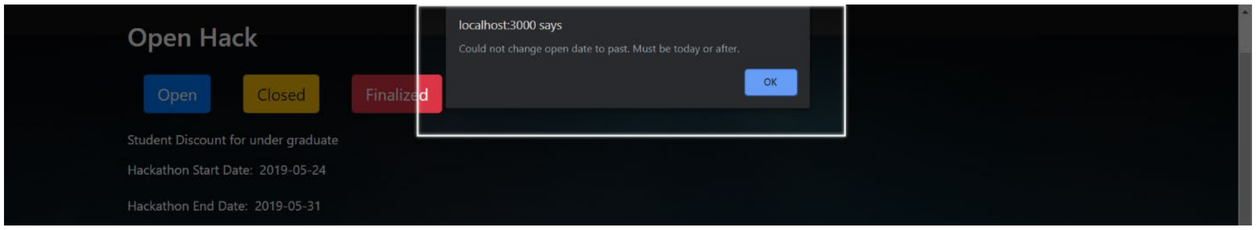
Hackathon Start Date: 2019-05-24

Hackathon End Date: 2019-05-31

New Start Date: 05/23/2019

New End Date: 05/25/2019

Validation like start date < end Date and date in past cannot be selected.



localhost:3000 says

Could not change open date to past. Must be today or after.

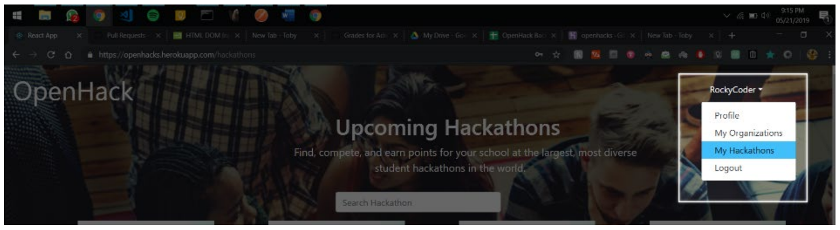
Student Discount for under graduate

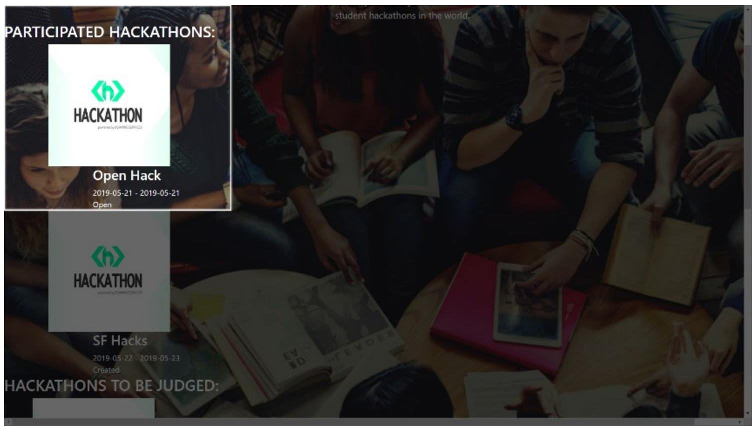
Hackathon Start Date: 2019-05-24

Hackathon End Date: 2019-05-31

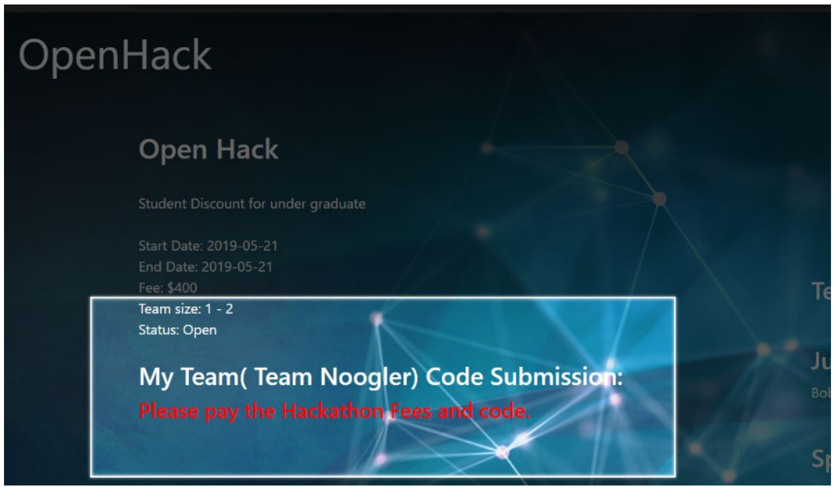
Code Submission

A non admin user can go and see his hackathons (participated and judged).


OpenHack
Upcoming Hackathons
Find, compete, and earn points for your school at the largest, most diverse student hackathons in the world.
Search Hackathon
RockyCoder
Profile
My Organizations
My Hackathons
Logout



If fee is not paid, the team submission cant be done.



OpenHack

Open Hack

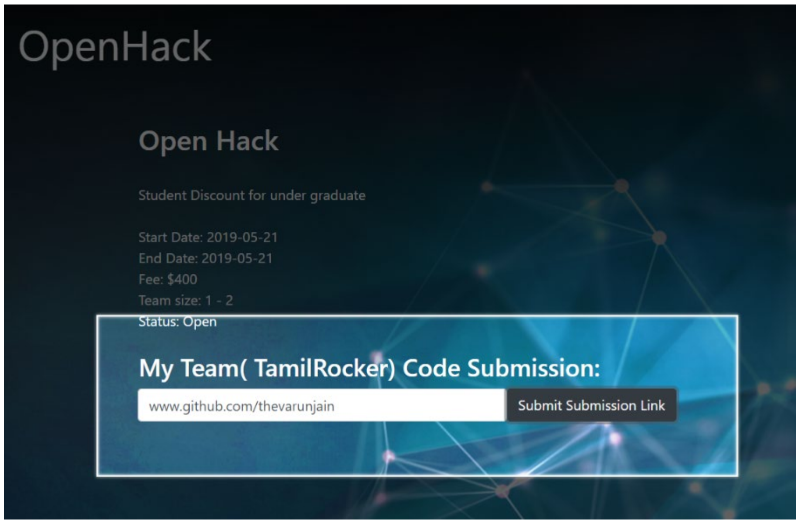
Student Discount for under graduate

Start Date: 2019-05-21
End Date: 2019-05-21
Fee: $400
Team size: 1 - 2
Status: Open

My Team(Team Noogler) Code Submission:

Please pay the Hackathon Fees and code.

Else if the fee is paid by all team member



OpenHack

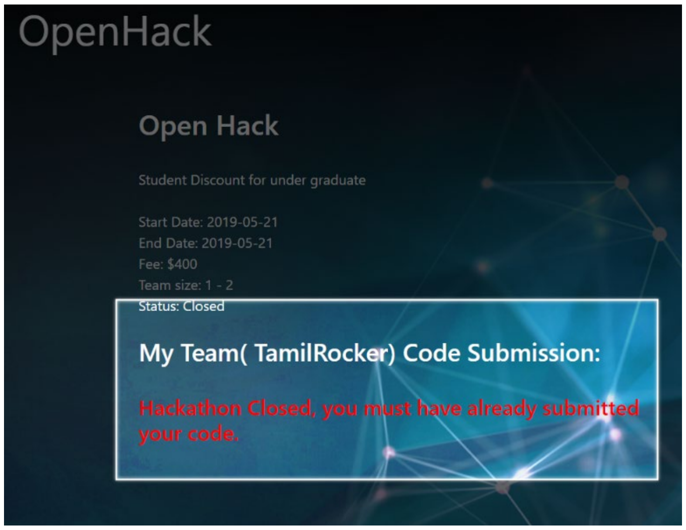
Open Hack

Student Discount for under graduate

Start Date: 2019-05-21
End Date: 2019-05-21
Fee: $400
Team size: 1 - 2
Status: Open

My Team(TamilRocker) Code Submission:

www.github.com/thevarunjain



OpenHack

Open Hack

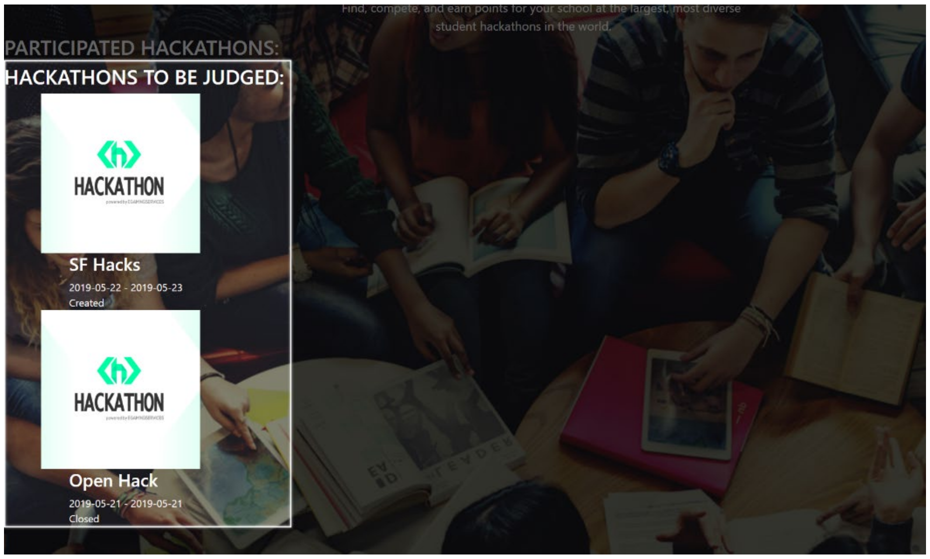
Student Discount for under graduate

Start Date: 2019-05-21
End Date: 2019-05-21
Fee: $400
Team size: 1 - 2
Status: Closed

My Team(TamilRocker) Code Submission:

Hackathon Closed, you must have already submitted your code.

Grading a team in a Hackathon

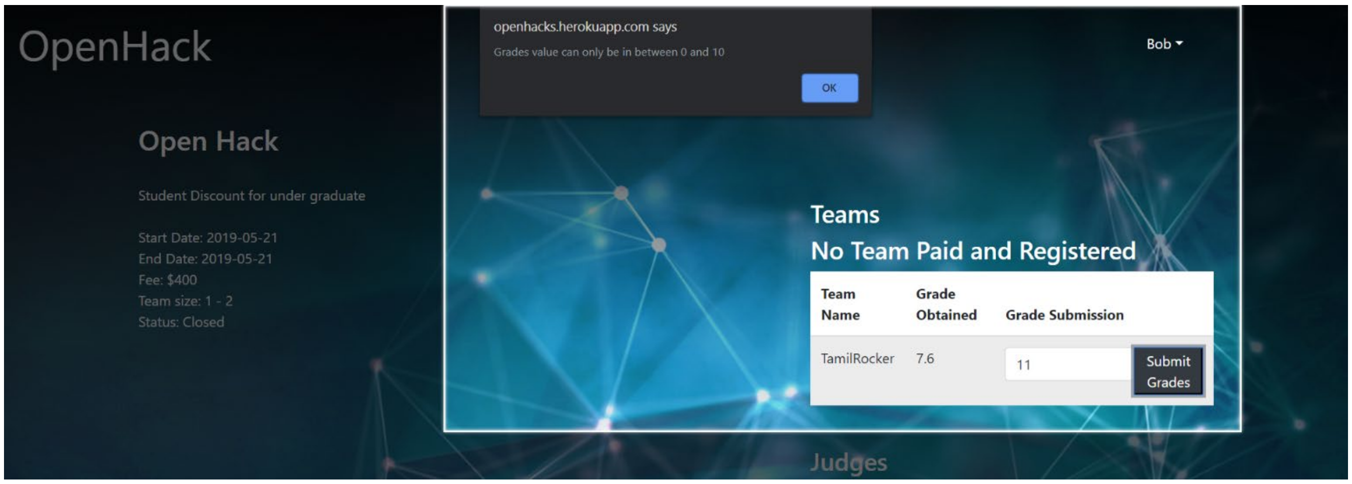

Find, compete, and earn points for your school at the largest, most diverse student hackathons in the world.

PARTICIPATED HACKATHONS:

HACKATHONS TO BE JUDGED:

SF Hacks
2019-05-22 - 2019-05-23
Created

Open Hack
2019-05-21 - 2019-05-21
Closed


OpenHack

openhacks.herokuapp.com says
Grades value can only be in between 0 and 10

Open Hack

Student Discount for under graduate

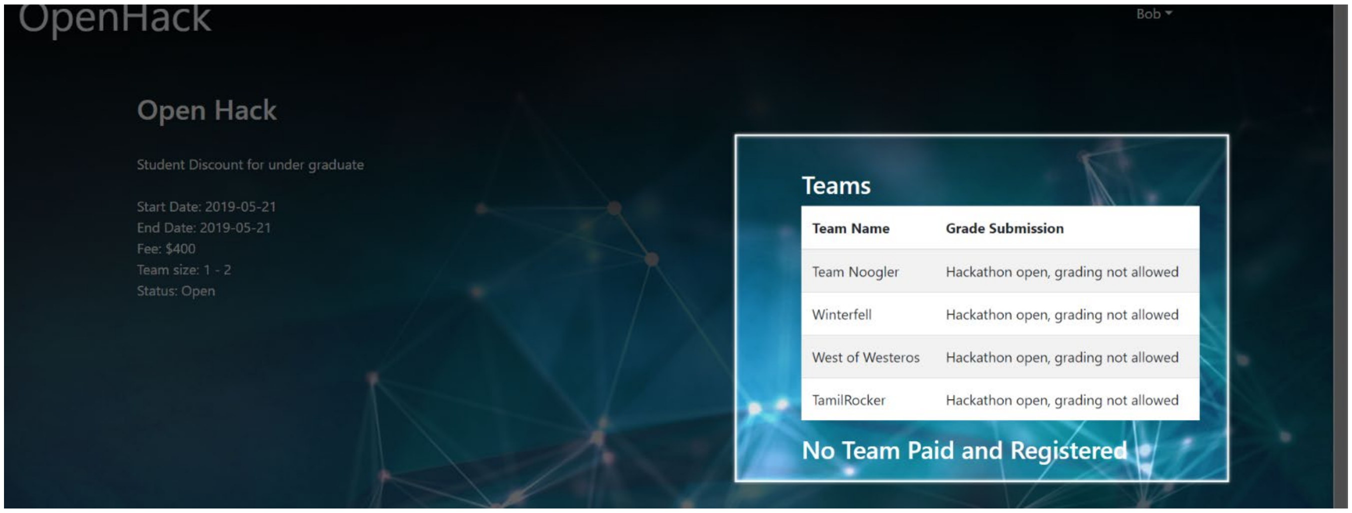
Start Date: 2019-05-21
End Date: 2019-05-21
Fee: $400
Team size: 1 - 2
Status: Closed

Teams

No Team Paid and Registered

Team Name	Grade Obtained	Grade Submission
TamilRocker	7.6	11

Judges


OpenHack

Open Hack

Student Discount for under graduate

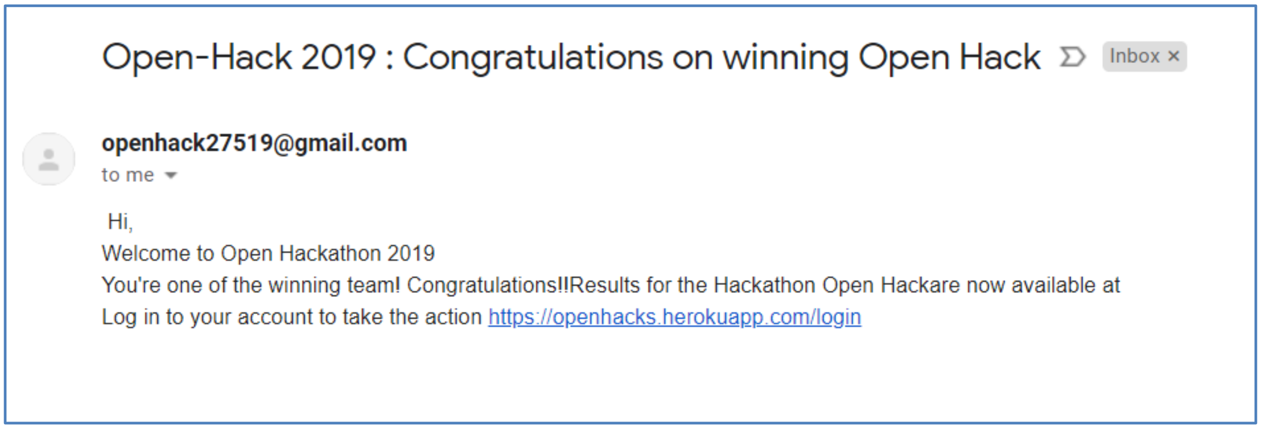
Start Date: 2019-05-21
End Date: 2019-05-21
Fee: $400
Team size: 1 - 2
Status: Open

Teams

Team Name	Grade Submission
Team Noogler	Hackathon open, grading not allowed
Winterfell	Hackathon open, grading not allowed
West of Westeros	Hackathon open, grading not allowed
TamilRocker	Hackathon open, grading not allowed

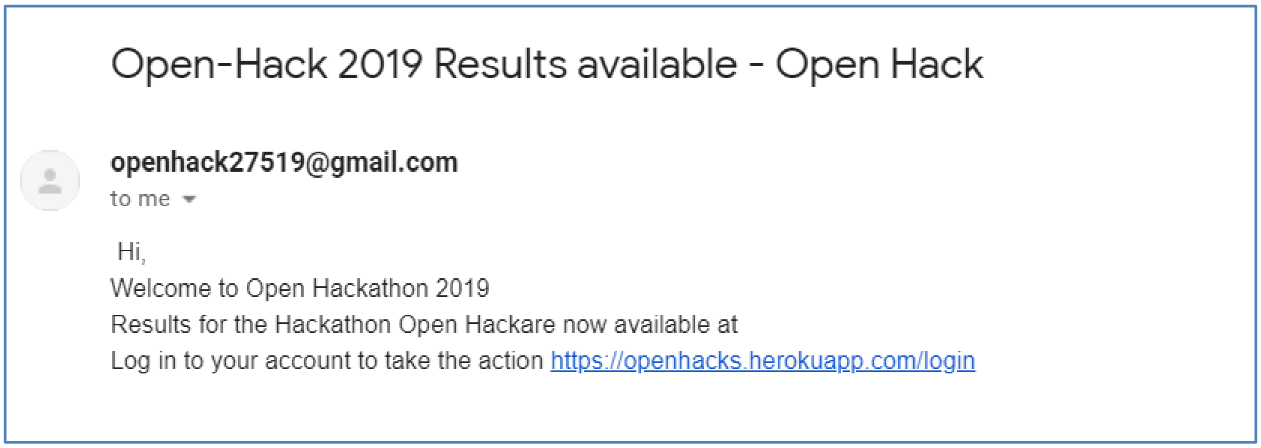
No Team Paid and Registered

On successful finalizing of Hackathon,


Open-Hack 2019 : Congratulations on winning Open Hack

openhack27519@gmail.com
to me

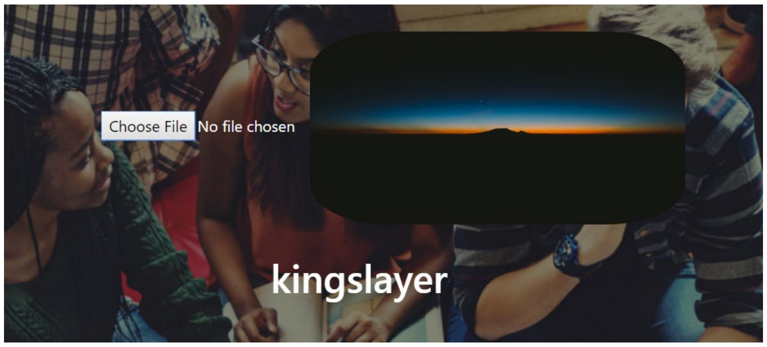
Hi,
Welcome to Open Hackathon 2019
You're one of the winning team! Congratulations!!Results for the Hackathon Open Hackare now available at
Log in to your account to take the action https://openhacks.herokuapp.com/login

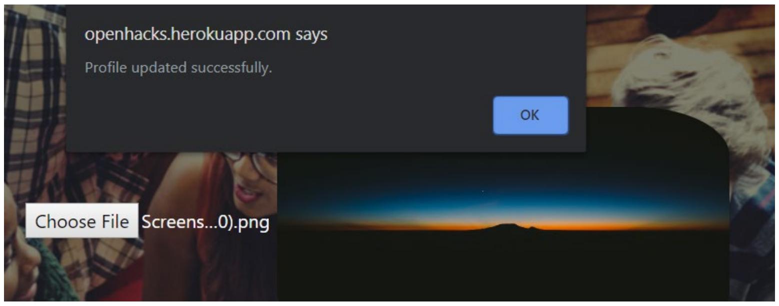

Open-Hack 2019 Results available - Open Hack

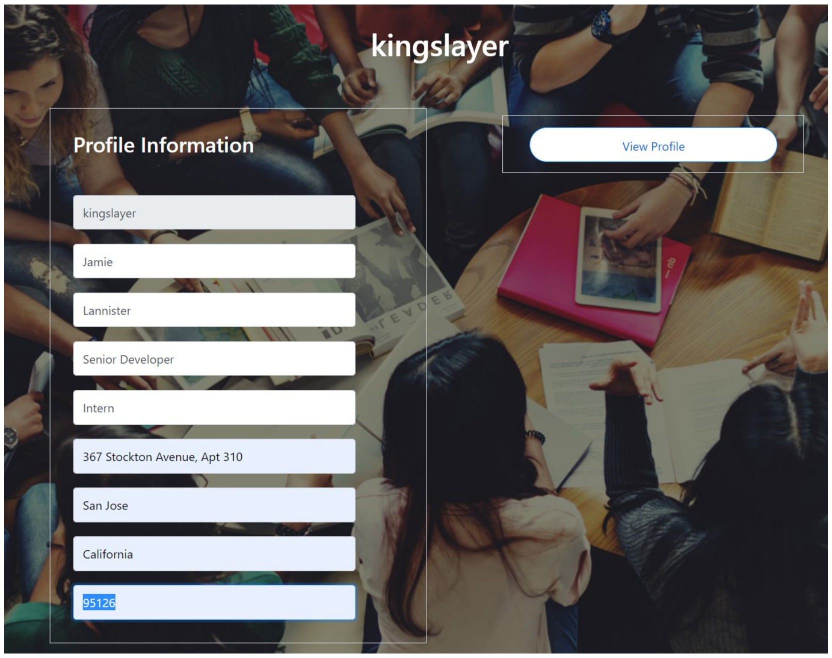
openhack27519@gmail.com
to me

Hi,
Welcome to Open Hackathon 2019
Results for the Hackathon Open Hackare now available at
Log in to your account to take the action https://openhacks.herokuapp.com/login

Updating a profile and profile photo




openhacks.herokuapp.com says
Profile updated successfully.



kingslayer

Profile Information

kingslayer

Jamie

Lannister

Senior Developer

Intern

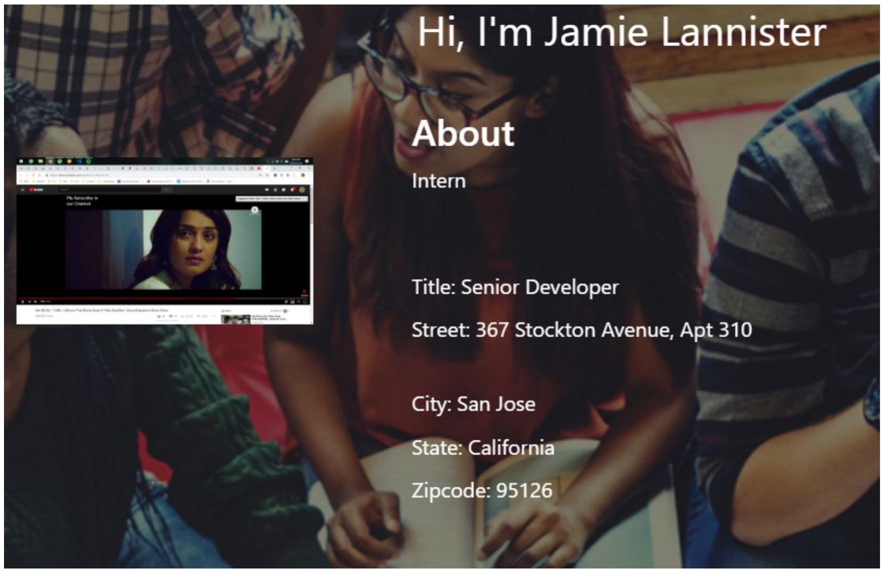
367 Stockton Avenue, Apt 310

San Jose

California

95126

Final profile after update



Hi, I'm Jamie Lannister

About

Intern

Title: Senior Developer

Street: 367 Stockton Avenue, Apt 310

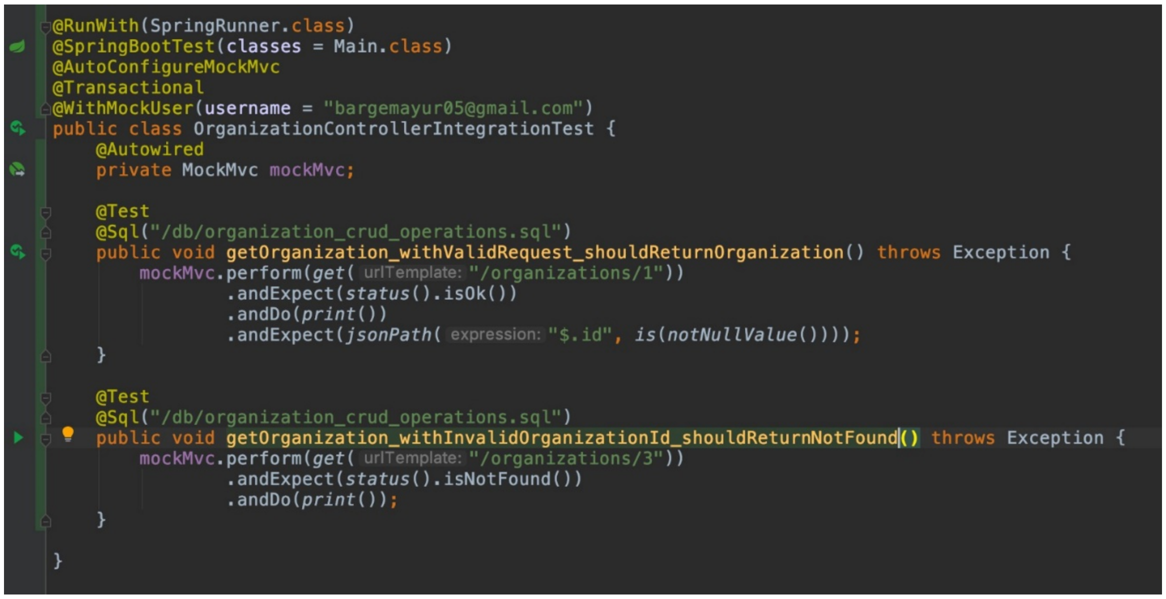
City: San Jose

State: California

Zipcode: 95126

Chapter 6: Test Plan and Results

We have used following methods for Testing
1. Manually tested all the API's using Postman initially
2. Integration test cases for controller are written and results are as below

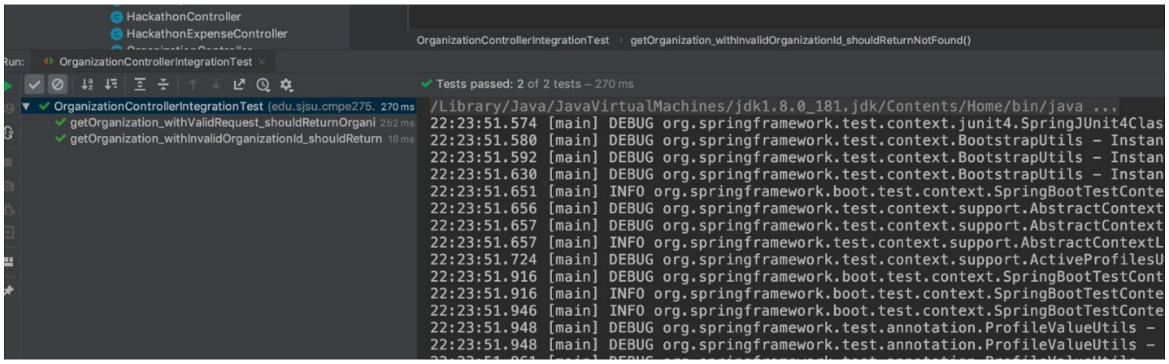


```java
@RunWith(SpringRunner.class)
@SpringBootTest(classes = Main.class)
@AutoConfigureMockMvc
@Transactional
@WithMockUser(username = "bargemayur05@gmail.com")
public class OrganizationControllerIntegrationTest {
    @Autowired
    private MockMvc mockMvc;

    @Test
    @Sql("/db/organization_crud_operations.sql")
    public void getOrganization_withValidRequest_shouldReturnOrganization() throws Exception {
        mockMvc.perform(get( urlTemplate: "/organizations/1"))
                .andExpect(status().isOk())
                .andDo(print())
                .andExpect(jsonPath( expression: "$.id", is(notNullValue())));
    }

    @Test
    @Sql("/db/organization_crud_operations.sql")
    public void getOrganization_withInvalidOrganizationId_shouldReturnNotFound() throws Exception {
        mockMvc.perform(get( urlTemplate: "/organizations/3"))
                .andExpect(status().isNotFound())
                .andDo(print());
    }

}
```



HackathonController
HackathonExpenseController
OrganizationControllerIntegrationTest › getOrganization_withInvalidOrganizationId_shouldReturnNotFound()
Run: OrganizationControllerIntegrationTest

```
✓ Tests passed: 2 of 2 tests – 270 ms

▼ ✓ OrganizationControllerIntegrationTest (edu.sjsu.cmpe275. 270 ms
    ✓ getOrganization_withValidRequest_shouldReturnOrgani 252 ms
    ✓ getOrganization_withInvalidOrganizationId_shouldReturn 18 ms

/Library/Java/JavaVirtualMachines/jdk1.8.0_181.jdk/Contents/Home/bin/java ...
22:23:51.574 [main] DEBUG org.springframework.test.context.junit4.SpringJUnit4Clas
22:23:51.580 [main] DEBUG org.springframework.test.context.BootstrapUtils - Instan
22:23:51.592 [main] DEBUG org.springframework.test.context.BootstrapUtils - Instan
22:23:51.630 [main] DEBUG org.springframework.test.context.BootstrapUtils - Instan
22:23:51.651 [main] INFO org.springframework.boot.test.context.SpringBootTestConte
22:23:51.656 [main] DEBUG org.springframework.test.context.support.AbstractContext
22:23:51.657 [main] DEBUG org.springframework.test.context.support.AbstractContext
22:23:51.657 [main] INFO org.springframework.test.context.support.AbstractContextL
22:23:51.724 [main] DEBUG org.springframework.test.context.support.ActiveProfilesU
22:23:51.916 [main] DEBUG org.springframework.boot.test.context.SpringBootTestCont
22:23:51.916 [main] INFO org.springframework.boot.test.context.SpringBootTestConte
22:23:51.946 [main] INFO org.springframework.boot.test.context.SpringBootTestConte
22:23:51.948 [main] DEBUG org.springframework.test.annotation.ProfileValueUtils -
22:23:51.948 [main] DEBUG org.springframework.test.annotation.ProfileValueUtils -
```

Chapter 7: Lessons Learned

1. Server application development using Spring boot allowed us for quick development cycle
2. Database migrations using Flyway helped us to manage database effectively
3. Using java libraries like Lombok helped us to bootstrap with Entity classes using builder
4. We have used Heroku for deployment and it is easy to use with GitHub for automated deployments
5. We were able to incorporate new features in backend code because of code modularity

Chapter 8: Future Work

1. Audit logging needs to be enabled for hackathon
2. Server-side application can be divided into different microservices to improve scalability
3. Testing and Production environment configurations can be made available so that application
   can be tested easily
4. HATEOAS project can be utilized for better REST representation
5. User interface can be more intuitive