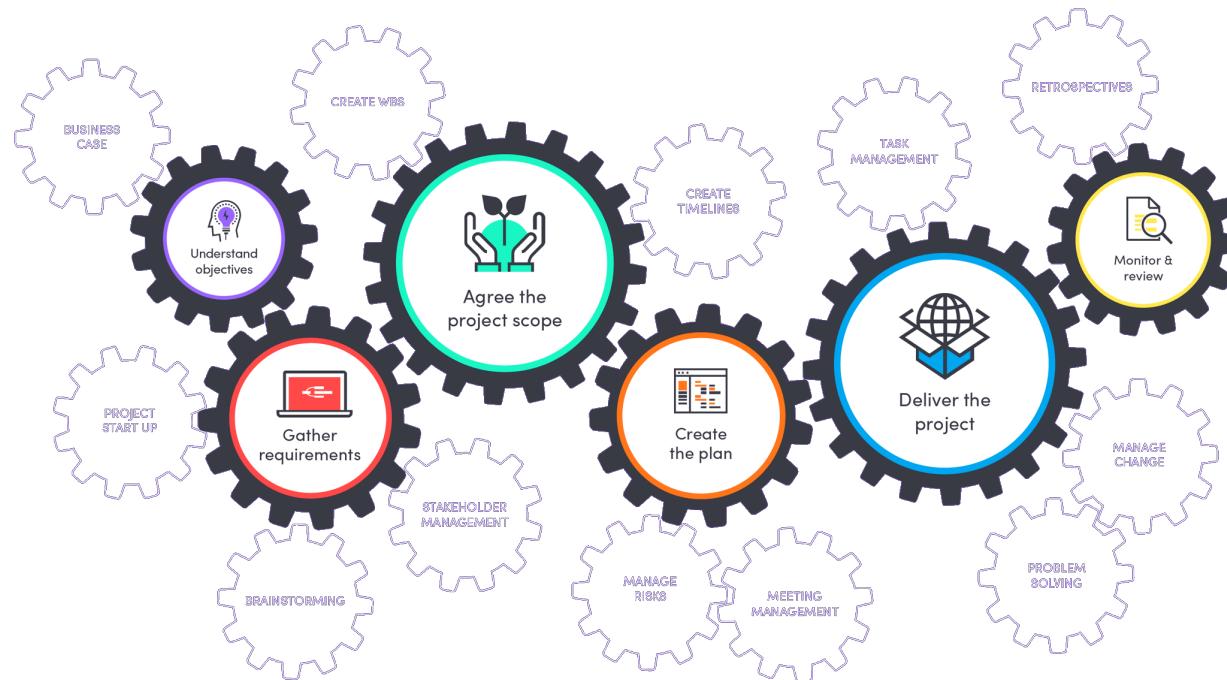


# VipraHub

## Project Final Submission

**CS5551: Advanced Software Engineering,  
Department of Computer Science Electrical Engineering,  
University of Missouri-Kansas City,  
Spring 2019.**



### Video:

<https://www.youtube.com/watch?v=PobY7pBp0eM&feature=youtu.be>

### Deployed URL:

<https://viprahub.herokuapp.com/>

### Git Hub:

<https://github.com/NehaNavgale/VipraHub>

### ZenHub:

<https://github.com/NehaNavgale/VipraHubworkspace/viprahub5c5b25c9ecb0076f3e899927/board?repos=168255028>

### Team Members, Class ID:

Chakra Pavan Kumar Kota, 13  
Naveena Madepally, 17  
Dharani Muli, 18  
Neha Navgale, 19  
(Team-3)

## **INDEX**

<b>S.NO</b>	<b>DESCRIPTION</b>
<b>1. PROJECT DEPLOYMENT</b>	
1.1	MOTIVATION
1.2	OBJECTIVE
1.3	WORKFLOW
1.4	DEVICE CONSTRAINTS
1.5	HOW TO USE THE SYSTEM
1.6	ERROR RECOGNITION AND HANDLING
1.7	DEPLOYMENT
<b>2. PROJECT MANAGEMENT REPORT</b>	
2.1	PROJECT MANAGEMENT REPORT
2.2	FINAL PROJECT EVALUATION
<b>3. PROJECT DEMO</b>	
<b>4. PROJECT PRESENTATION SLIDES</b>	
<b>5. PROJECT PROPOSAL AND PLAN</b>	
<b>6. FIRST INCREMENT REPORT</b>	
<b>7. SECOND INCREMENT REPORT</b>	
<b>8. THIRD INCREMENT REPORT</b>	
<b>9. FOURTH INCREMENT REPORT</b>	
<b>10. REFERENCES</b>	
<b>11. ACKNOWLEDGEMENT STATEMENTS</b>	

# 1. PROJECT DEPLOYMENT

## 1.1 MOTIVATION

There are many model repository sites and hosting sites available but there is no one stop place for the end user to have both the features. In order to overcome this problem, we are motivated to create a single application with high end features like share, test, deploy and visualize the output online.

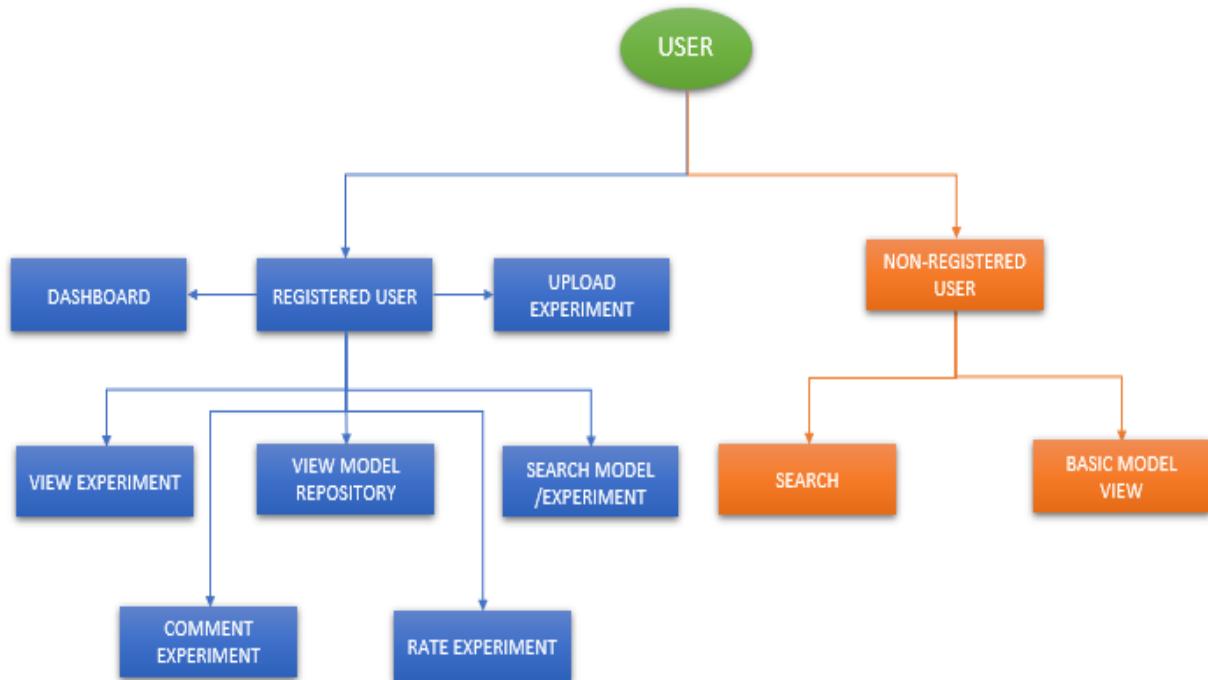
## 1.2 OBJECTIVE

Our objective is to design an open source application with maximum features which promote users to do reproducible research, manage all their models at one place. Moreover, this will also provide great visualization and exposure of contributor's work.

## 1.3 DEVICE CONSTRAINTS:

This is Web application which runs on any browser.

## 1.4 WORKFLOW:



## 1.5 HOW TO USE THE SYSTEM

### Home page:

This is the landing page of the application. Once the user hit [Viprahub](#) URL, home page will open up. The page contains following features:

1. Option to Sign Up for new account
2. Option to Login if already have account
3. Search for all the models uploaded so far.
4. Check top 3 models by accuracy.
5. Able to check top 3 models of each category by clicking on category name.
6. View the features that Viprahub provides.

## **Registration Page:**

In order to create new account, click on Sign up button on header of home page. Registration page will be displayed.

All the fields on Registration page are mandatory. Fill First Name, Last Name, Email ID, Password and confirm password fields and click on Register button.

The account will get created and user will be redirected to Login Page.

## **Login Page:**

If user already have account, they can Login by clicking on Login button available on header of home page.

Provide valid User Name (Email id) and Password and click on Login button.

## **User Dashboard Page:**

This the dashboard page of user. User will be able to view following details about his uploaded models.

1. Models table will display all the models uploaded by user and its metadata.
2. Total number of models uploaded
3. How many downloads are made for his models

Log out option is provided on top right corner. Once user clicks on Log out button, he will be redirected to Login Page.

User have option to upload new models/experiments by clicking on “Click me to upload your model”

## **Upload:**

On click on “Click me to upload your model” button from User Dashboard page, a dialog box will open up to upload experiment of any model.

1. Select category of model from category dropdown.
2. Give model name
3. Give experiment name.
4. Upload relevant files of models by clicking choose files button.
5. User can select multiple files from local system to upload

6. Once files are selected from local system, click on open. Names of all the files selected will be visible on dialog box.
7. Remove button is also provided in order to remove any file from upload list.
8. Click on Upload All option. Upload will start and progress of uploads will be shown in progress bar.
9. Once all the files are uploaded click on close(X).

## **My Models:**

This page will provide details about models in folder structure. This page is accessible by clicking on 'My models' link on User Dashboard page.

The page will show all the models uploaded by user. Once user click on any model name, it will display all the experiments uploaded inside model. Clicking on experiment name will navigate to Model dashboard page.

## **Search Page:**

This page provides a feature to search for all the models uploaded so far by all the user. Search box provided on header is a global search box.

Enter any keyword in search box, the results matching the keyword entered will be displayed on search page. User can view the search results in two views:

1. Grid View
2. Table view

Views can be changed by clicking on Grid and table icon provided on the top right of search results.

User can again filter the search results by selecting Category from the left menu.

User can sort the results based on accuracy and year. Sorting option is provided on the top right of the page.

Click on any of the search result, user will be navigated to model dashboard page.

## **Model Dashboard Page:**

This page will display metadata of selected model. The page has following feature:

1. View Overview of the model
2. Publication details (If any)
3. Rate Models
4. Provide Comments on Model
5. Download Model

## **Download:**

Click on Download button from Model Dashboard page. All the files of model will be downloaded in a zipped format on local system.

### **Rate:**

Rating option is provided on model dashboard page. Click on Rate button it will open a pop up to rate the model on 1- 5 scale.

Please note: if logged in user is an owner of the model then he cannot rate his own model. The system has validation to restrict user to rate his own model.

### **Comments:**

Comments tab on model dashboard page provides feature to provide comments related to model and view the previous comments.

Enter comment in the textbox provided and click on Comment button. The comment will be posted and will be visible on comment tab.

## **1.6 ERROR RECOGNITION AND HANDLING**

We have added all the possible validations on each and every module.

### **Registration Page:**

- Text fields cannot be empty.
- Password and confirm password must match
- Email ID must be unique

### **Login Page:**

- Fields cannot be empty
- Credentials must match to the saved credentials on the database.

### **Upload Model:**

- Category should be selected
- Model and experiment name should not be empty
- Experiment name should be unique.

### **Rate Model:**

- User should not be able to rate his own model.

## **1.7 DEPLOYMENT:**

Viprahub is deployed in two instances in Heroku cloud provider. One instance holds the frontend part of the application. The other is responsible for providing backend services and interacting with mongodb atlas.

## Front end: (Main Application)

<https://viprahub.herokuapp.com/>

The screenshot shows the Heroku dashboard for the 'viprahub' application. At the top, there's a navigation bar with the Heroku logo, a search bar labeled 'Jump to Favorites, Apps, Pipelines, Spaces...', and a user profile icon. Below the navigation is a header with a user icon, the app name 'viprahub', and GitHub information ('GitHub NehaNavgale/VipraHub master'). To the right are buttons for 'Open app' and 'More'. The main content area has tabs for 'Overview', 'Resources', 'Deploy', 'Metrics', 'Activity', 'Access', and 'Settings'. Under 'Overview', sections include 'Installed add-ons (\$0.00/month)' (no add-ons), 'Dyno formation (\$0.00/month)' (using free dynos, command: 'web npm start'), and 'Collaborator activity' (nehanavgale0604@gmail.com, 24 deploys). On the right, the 'Latest activity' log shows several deployment and build logs from May 7, 2023, by nehanavgale0604@gmail.com.

Event	Time	Details
Deployed	May 7 at 5:47 PM	v26 · afbf1906
Build succeeded	May 7 at 5:44 PM	View build log
Deployed	May 7 at 5:54 PM	v25 · 070b70f8
Build succeeded	May 7 at 5:51 PM	View build log
Deployed	May 7 at 5:47 PM	v24 · a656694e
Build succeeded	May 7 at 5:45 PM	View build log

## Back end services:

<https://viprahubbackend.herokuapp.com/>

The screenshot shows the Heroku dashboard for the 'viprahubbackend' application. It has a similar layout to the first dashboard, with a navigation bar, user profile, and GitHub info. The 'Overview' section shows 'Installed add-ons (\$0.00/month)' (no add-ons) and 'Dyno formation (\$0.00/month)' (using free dynos, command: 'web node Server.js'). The 'Collaborator activity' section shows nehanavgale0604@gmail.com with 22 deploys. The 'Latest activity' log on the right shows deployment and build logs from May 7, 2023, by nehanavgale0604@gmail.com.

Event	Time	Details
Deployed	May 7 at 5:09 PM	v24 · 51168231
Build succeeded	May 7 at 5:09 PM	View build log
Deployed	May 7 at 4:15 PM	v23 · b0ffe204
Build succeeded	May 7 at 4:14 PM	View build log
Deployed	May 7 at 9:40 AM	v22 · cb2842f6
Build succeeded	May 7 at 9:39 AM	View build log

# 2 PROJECT MANAGEMENT

## 2.1 PROJECT MANAGEMENT REPORT:

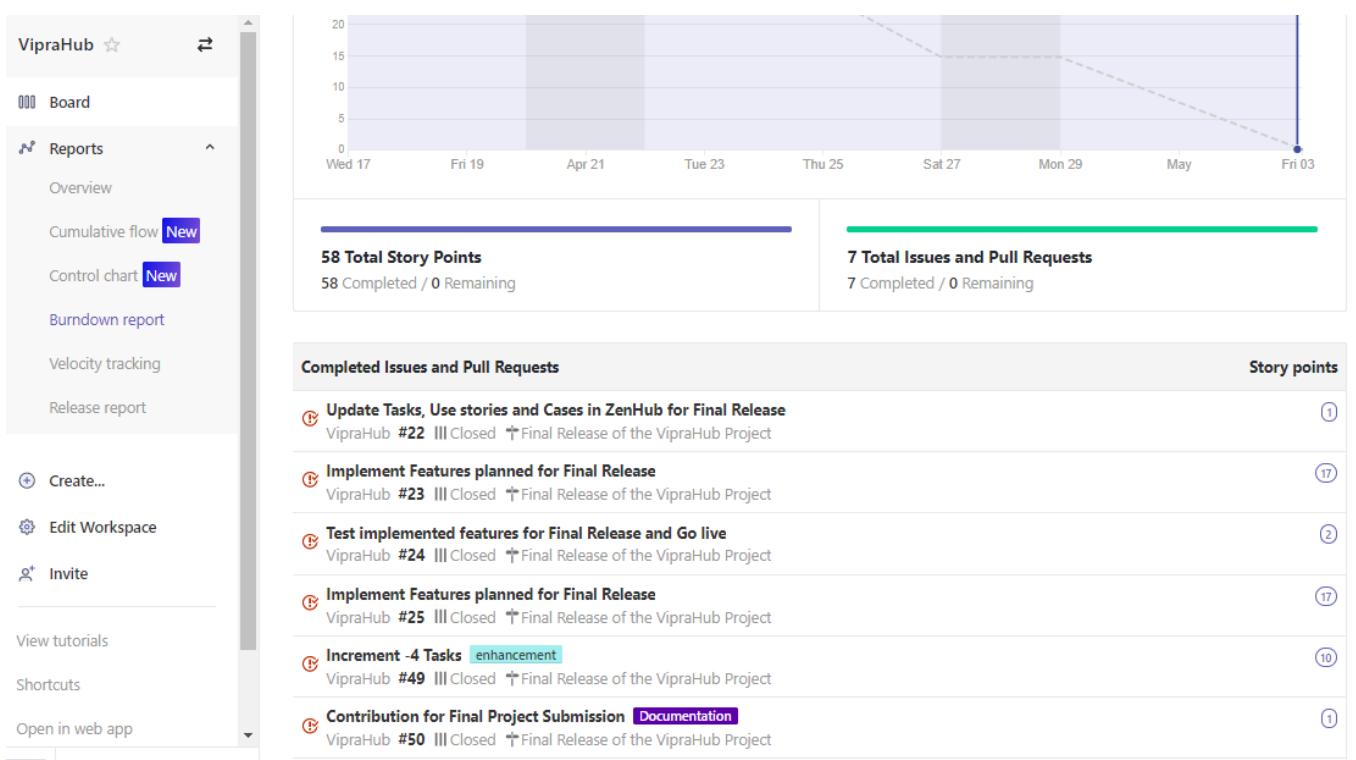
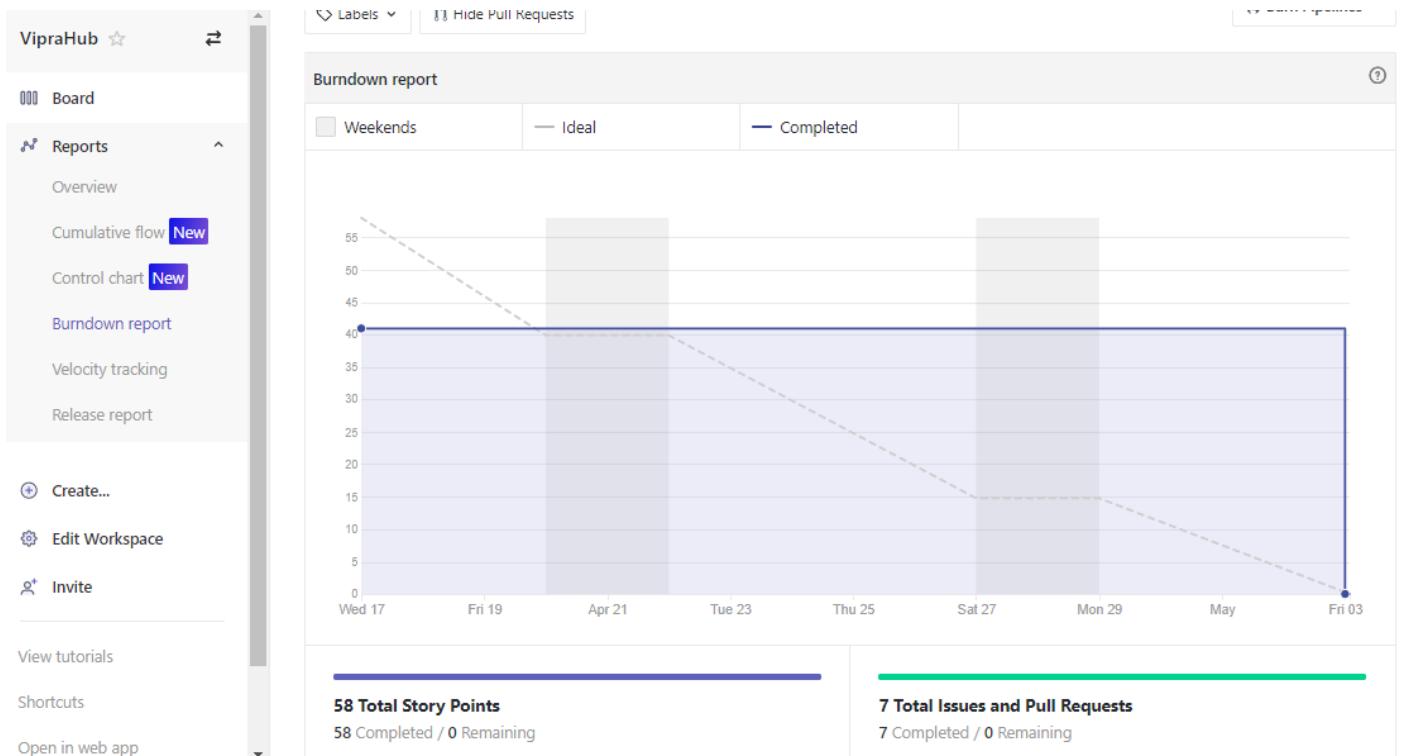
The entire project was managed, well planned and executed with the help of professional project management tools such as Zenhub.

**The project has been executed in the following sequence.**

- 1) Several meetings are conducted in the initial phase to understand the requirements and getting the resolutions for clarifications with Project Manager.
- 2) Later, a compiled initial set of features were identified and presented at the Project Proposal meeting.
- 3) All the tasks were loaded into Zenhub repository and corresponding points were assigned for each task to be executed.
- 4) The project execution couldn't have been smoother without the Stand Up meetings we had.
- 5) Each tasks are continuously monitored and the risks that raised due to the busy schedules are mitigated using the help of Zenhub
- 6) The closure, release plans were jotted down and closed in the Zenhub.

**Screenshots:**

The screenshot shows the ZenHub interface for the repository 'NehaNavgale / VipraHub'. The top navigation bar includes 'Pull requests', 'Issues', 'Marketplace', and 'Explore'. The main area displays a 'Board' view with four columns: 'Milestones' (0 Issues), 'Review/QA' (0 Issues), 'Done' (0 Issues), and 'Closed' (50+ Issues). The 'Closed' column contains several items, each with a user icon, a title like 'VipraHub #51 Increment-1 Tasks', and a description. The left sidebar provides navigation links for 'Board', 'Reports', 'Create...', 'Edit Workspace', 'Invite', 'View tutorials', 'Shortcuts', 'Open in web app', 'Support and training', and a 'Changelog' section for user 'pavankotas'. The bottom left corner shows the user's profile picture and name 'pavankotas'.



## Task Allocation Strategy:

### Design:

Everyone has an equal contribution in the design phase which is the crucial part of the entire project execution.



### Implementation:

Based on the project phase everyone has their major contributions in terms of the allocated modules. Each module has been initiated, enhanced by all the team mates in a round robin fashion. This helped everyone in the team to gain grip on the entire project instead of having a tunnel view.



### Testing:

The testing phase which is one of crucial parts of project execution has been performed in an incremental way. Every developer took the responsibility of testing the modules that he/she has taken up. However, the integration testing has been owned by Dharani and Naveena.



### Deployment:

The application is private until it is released. Deployment allows the application to be accessible from anywhere. So, we have spent time in hosting the application both frontend and backend services into Heroku cloud provider. There are lot of challenges

we encountered in setting the environments, but at the end we were able to succeed as a team. Everyone in the team is equally contributed. However, Neha and Pavan acted as owners for this phase.



## 2.2 FINAL PROJECT EVALUATION

- 1) How well your project satisfies your original requirement specifications?

The project completely aligns with the Software Requirements Specifications (SRS). However, the requirements have few suggestions, change requests which usually happens in agile process. We were able to accommodate those changes and stick to the timeline. In a scale of 1 to 10, we would rate ourselves as 9.5 in terms of fulfilling the project requirements.

- 2) Were you satisfied with your design process?

Yes, the final design of the project that we have now is the result of multiple iterations of verifications and validations. The DB design is very flexible, it can accommodate changes without disturbing the existing functionalities. We can confidently say the design is very apt when considering large scale applications like the project.

- 3) How helpful was the agile process? How would you do the agile process next time?

Agile is the buzz word in the current software industry. We were able to understand what exactly it is and how it works by undertaking the project. In the past we had experience working with this model. But, the major learning here is despite having other course works the process motivated, pushed us in bringing the final product. We enjoyed doing the process in an agile manner. In future, with all the takeaways we had here we would use the agile model in building large scale products.

- 4) Did you stick to your project plan schedule?

Yes, we were able to close all the tasks on time every time during all the four increments. We had tough situations sometimes (Tasks are lagging, Pressures from another course works). But we as a team spent extra time than planned to get the things closed. The outcome is ultimately achieved.

- 5) What was the real management structure within your group?

Our team is structured in a way we had

- 1) Scrum Master/ Team Lead
- 2) Scrum Team

However, we had a responsible owner for each major division. Owners include

- 1) Presentation Owner
- 2) Documentation Owner
- 3) Design Owner
- 4) Deployment owner

- 6) Does it bear any resemblance to structure that you had planned?

Yes, It was as planned.

- 7) Did you have any problems getting each member to do his/her share of the work?

Fortunately, we never had any problem with in our team, we as a team collaborated very well and the success of our project is all because of whole team.

- 8) Do you have any suggestion on how this could have been handled better?

If such is a scenario, we would try spending our initial time in understanding the problem. The problem can be technical deficiency, behavioral problem.

The technical deficiency can be resolved by providing proper knowledge transfer sessions.

Behavioral problems can still be addressed by having one on one discussions, providing help to the team mate in directing the destination.

- 9) Discuss what you might have done differently if this were a real-world project?

We have considered this project as a real-world project. We have few enhancements that we have in our mind that we couldn't do considering the time factor. If it was a real-world project we would do enhancements to this project in terms of performance, optimizations.

- 10) Any recommendations for next year?

To be honest, our team is having experienced professionals. We were able to build the application using the previous experiences. I feel some guidance needs to be given to students who might hesitate even in asking the right questions related to project. Ultimately, everyone should succeed. Projects are great way for students to learn.

### **3. PROJECT DEMO:**

Please find our demo video in the given below link:

<https://www.youtube.com/watch?v=PobY7pBpOeM&feature=youtu.be>

## 4. PROJECT PRESENTATION SLIDES



“

## Objective:

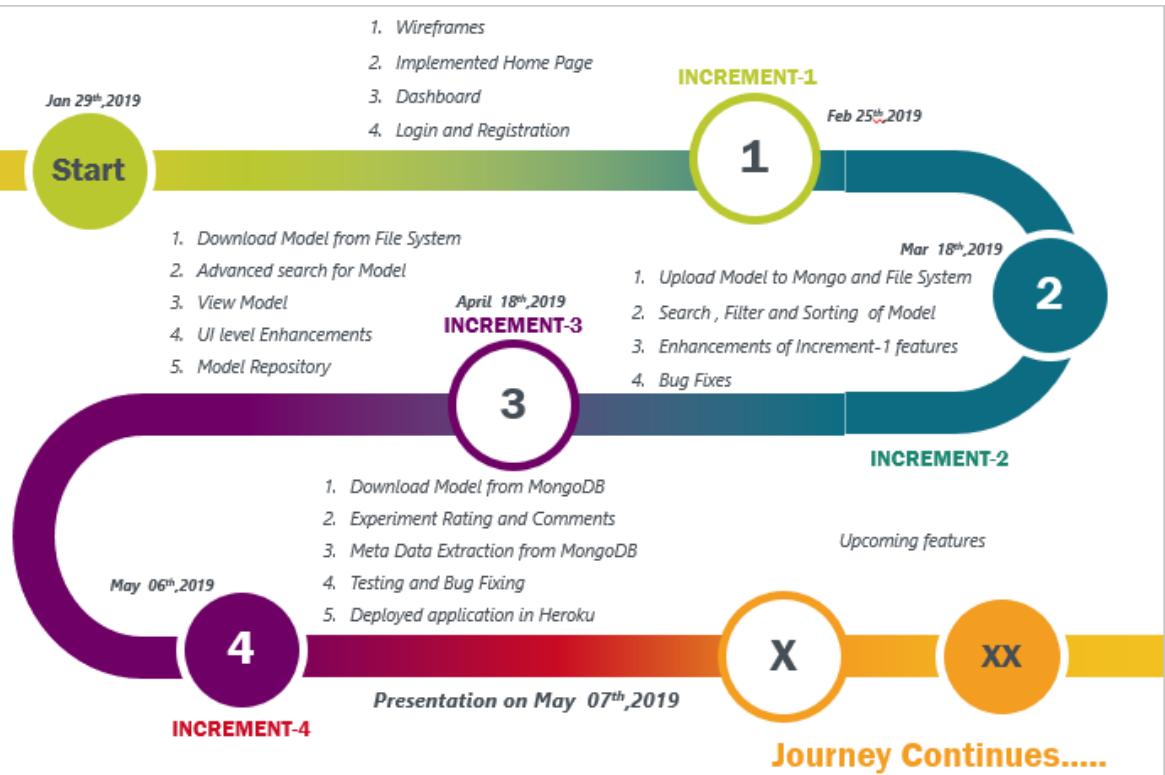
Our objective is to design an open source platform with maximum features which promote users to do reproducible research, manage all their models at one place. Moreover, this will also provide great visualization and exposure of contributor's work.

2

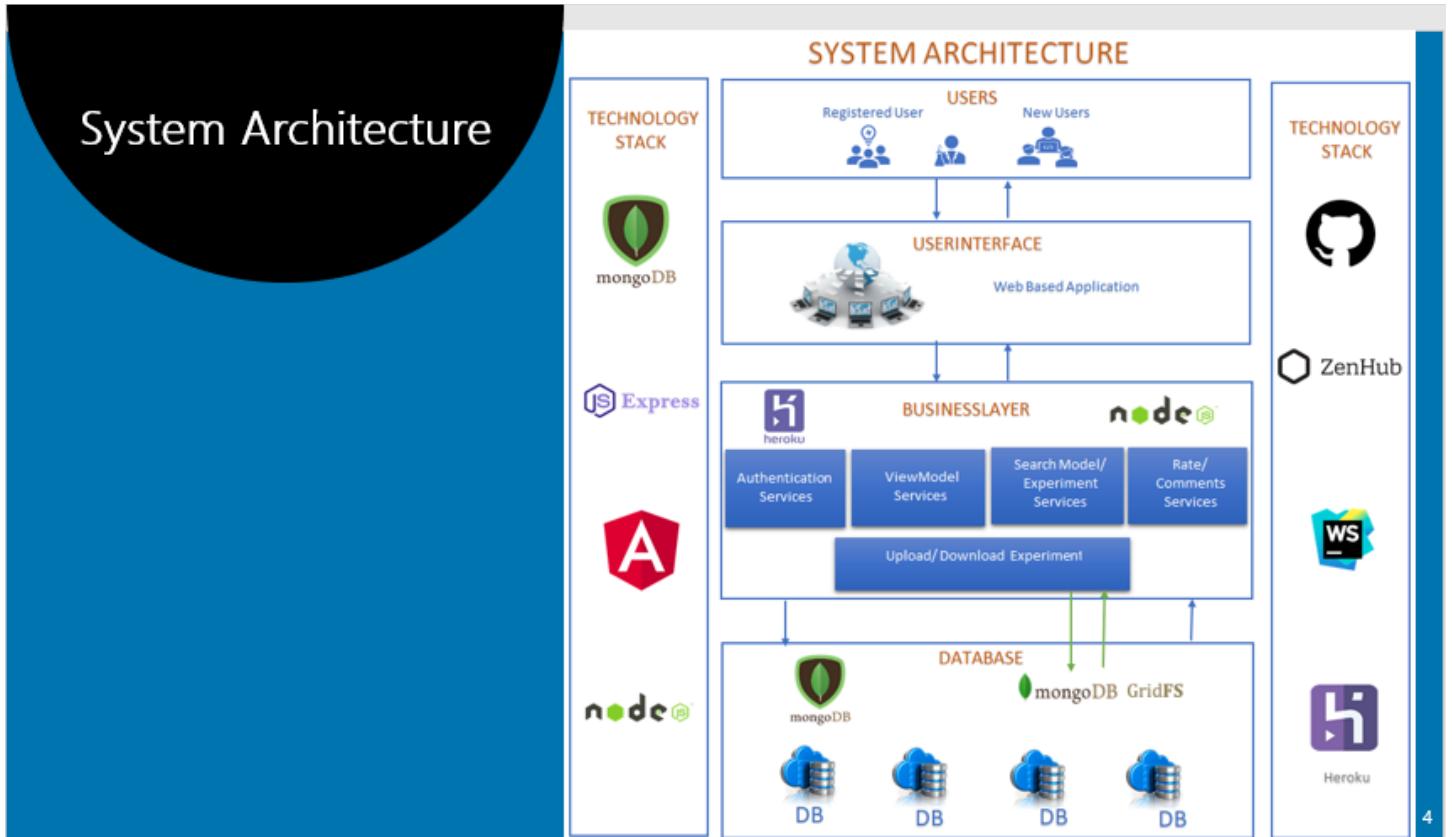
## PROJECT JOURNEY

VipraHub

Feb 15<sup>th</sup> 2019 to  
May 7<sup>th</sup> 2019



# System Architecture



VipraHub

Search Models

Search

Sign Up Log In

# Hub of Deep Learning Models. Deploy, Test and Visualize your models

Get Started



VipraHub

Search Models

Search

Sign Up Log In

## Browse by Categories

Image Classification

Text and Language

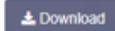
Sound and Music

Fairness and Bias

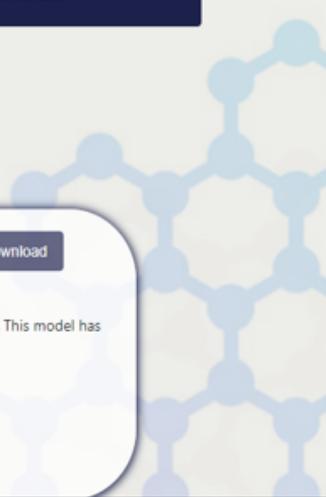
Pattern Recognition



**model1**

 Download

The model is published by Author [dharani@gmail.com](mailto:dharani@gmail.com) in the year . This model has following metadata:  
**Accuracy** : 0.46875  
**Optimizer** : Adam  
**Framework** : keras 2.2.2



VipraHub x +

https://viprahub.herokuapp.com/#/landing

Search Models Search

Sign Up Log In

### VipraHub



The model is published by Author [dharani@gmail.com](mailto:dharani@gmail.com) in the year . This model has following metadata:  
**Accuracy** : 0.46875  
**Optimizer** : Adam  
**Framework** : keras 2.2.2

### Text and Language Model



The model is published by Author [dharani@gmail.com](mailto:dharani@gmail.com) in the year . This model has following metadata:  
**Accuracy** : 0.59375  
**Optimizer** : Adam  
**Framework** : keras 2.2.2

Download

### Pattern Recognition



Download

<https://viprahub.herokuapp.com/#/landing>

# VipraHub

Search Models  Search

Sign Up Log In

Browse for more models

## Why VipraHub?

- Host your model**
- Test your model**
- Visualization**
- Search other model**
- Deploy your model online**
- Test your model** Test your model on VipraHub and allow others to test and rate your model
- Great Visualization for models and its metadata**
- Browse models from a list of different kinds of deep learning categories and sort it based on accuracy and rating**

Vipra Hub is managed by developers of University of Missouri Kansas City. Contact us for sharing idea or suggestions

Copyright © team 3

VipraHub

Sign in to VipraHub

User Name

Password

Login

Don't have an account? [Sign up](#)

A screenshot of the VipraHub dashboard. The left sidebar has a dark blue header with the "VipraHub" logo and a "Dashboard" icon. Below it, under the "MODELS" section, are links for "Search Models" and "My Models". The main content area has a light gray header with a search bar containing the placeholder "Search Models" and a user profile icon with the email "dharani@gmail.com". In the center, there are two teal-colored boxes. The left box, titled "Click me to upload your Model", shows a large number "3" and the text "Models uploaded by you" above a line chart with three data points. The right box, also titled "3", shows the text "Downloaded by others" above a line chart with four data points. Below these boxes is a table titled "Models" with columns: Model Name, Accuracy, Optimizer, Framework, and Download Count. The table contains three rows of data:

Model Name	Accuracy	Optimizer	Framework	Download Count
model1	0.46875	Adam	keras 2.2.2	1
Text and Language Model	0.59375	Adam	keras 2.2.2	0
FairnesMdel	0.4375	Adam	keras 2.2.2	2

Viprahub https://viprahub.herokuapp.com/#/theme/search

VipraHub Search Models dharani@gmail.com

Dashboard MODELS Search Models My Models

## Search Results

Filter By Category

- Image Classification
- Text and Language
- Sound and Music
- Fairness and Bias
- Pattern Recognition



### FairnesMdel

The model is published by Author [dharani@gmail.com](#) in the year . This model has following metadata:  
**Accuracy :** 0.4375  
**Optimizer :** Adam  
**Framework :** keras 2.2.2



### model1

The model is published by Author [dharani@gmail.com](#) in the year . This model has following metadata:  
**Accuracy :** 0.46875  
**Optimizer :** Adam  
**Framework :** keras 2.2.2

Sort By

Viprahub https://viprahub.herokuapp.com/#/theme/search

Data | Atlas: MongoDB Atlas

VipraHub Search Models pavankotas@gmail.com

Dashboard MODELS Search Models My Models

## Search Results

Filter By Category

- Image Classification
- Text and Language
- Sound and Music
- Fairness and Bias
- Pattern Recognition

Model Name	Author	Accuracy	Optimizer	Framework
FairnesMdel	dharani@gmail.com	0.4375	Adam	keras 2.2.2
model1	dharani@gmail.com	0.46875	Adam	keras 2.2.2
Text and Language Model	dharani@gmail.com	0.59375	Adam	keras 2.2.2
Sound Model	nehanavgale0604@gmail.com	0.974	Adam	keras 2.2.4
Pattern Recognition	nehanavgale0604@gmail.com	0.978	Adam	keras 2.2.4
Model4	nehanavgale0604@gmail.com	0.997	Adam	keras 2.2.4

VipraHub Data | Atlas: MongoDB Atlas https://viprahub.herokuapp.com/#/dashboard

VipraHub Fairn dharani@gmail.com

Dashboard MODELS Search Models My Models

Click me to upload your Model

**Upload Files**

Category ID: Model name: Experiment:

Choose Files No file chosen

Upload queue File(s) Selected: 0

Name	Size	Progress	Status
FairnessModel	0.4375	Adam	Keras 2.2.2

Total progress:

Upload all  Cancel all  Remove all

Show all X

VipraHub Data | Atlas: MongoDB Atlas https://viprahub.herokuapp.com/#/dashboard

VipraHub Fairn dharani@gmail.com

Dashboard MODELS Search Models My Models

Click me to upload your Model

**Upload Files**

Category ID: Model name: Experiment:

Choose Files No file chosen

Upload queue File(s) Selected: 0

Name	Size	Progress	Status
FairnessModel	0.4375	Adam	Keras 2.2.2

Total progress:

Upload all  Cancel all  Remove all

Show all X

VipraHub

Search Models

Upload Files

Category ID: Image Classification Model name: model10 Experiment: Model10Exp

Upload queue

File(s) Selected: 5

Name	Size	Progress	Status
CNN011019-221140_accuracy.jpg	0.018 MB	<div style="width: 100%;"> </div>	✓ <button>Remove</button>
CNN011019-221140_architecture.jpg	0.025 MB	<div style="width: 100%;"> </div>	✓ <button>Remove</button>
CNN011019-221140_loss.jpg	0.018 MB	<div style="width: 100%;"> </div>	✓ <button>Remove</button>
CNN011019-221140_metadata.txt	0.00 MB	<div style="width: 100%;"> </div>	✓ <button>Remove</button>
CNN011019-221140_model.h5	9.34 MB	<div style="width: 100%;"> </div>	✓ <button>Remove</button>

Total progress:

Upload all Cancel all Remove all

Load Count

Viprahub

Data | Atlas: MongoDB Atlas

https://viprahub.herokuapp.com/#/theme/colors

FairnesMdel ★★★★★

Fairness Exp Rate

Download

OVERVIEW PUBLICATION ARCHITECTURE TEST COMMENTS

**Model Name :** FairnesMdel  
**Experiment :** Fairness Exp  
**Author :** dharani@gmail.com  
**Category :** 5c8df10a1c9d4400002155be  
**Framework :** keras 2.2.2  
**Size :** 9793.952 kilobytes  
**Input Tensors :** (None, 64, 64, 3)  
**Output Tensors :** (None, 1)  
**Optimizer :** Adam  
**Loss Function :** binary\_crossentropy  
**Accuracy Value :** 0.4375  
**Loss Value :** 0.7406877279281616  
**Year :** N/A

The screenshot shows a web browser window with the URL <https://viprahub.herokuapp.com/#/theme/colors>. The page displays a 'Text and Language Model' with a five-star rating. The 'OVERVIEW' tab is selected, showing detailed model information:

Model Name	: Text and Language Model
Experiment	: Text and Language Experiment 1
Author	: dharani@gmail.com
Category	: 5c8df0d81c9d4400002155bc
Framework	: keras 2.2.2
Size	: 9793.952 kilobytes
Input Tensors	: (None, 64, 64, 3)
Output Tensors	: (None, 1)
Optimizer	: Adam
Loss Function	: binary_crossentropy
Accuracy Value	: 0.59375
Loss Value	: 0.7149171829223633
Year	: N/A

A red arrow points from the taskbar to the file icon in the system tray, indicating where the downloaded file is located.

The screenshot shows a Windows taskbar with the file 'Text and Language....zip' pinned. The taskbar also includes icons for File Explorer, Edge, and other system applications. Below the taskbar, the VipraHub interface is visible, showing the 'Your Uploaded Models' section with four uploaded models: 'model1', 'Model2', 'Text and Language Model', and 'FairnesMdel'. A red arrow points from the taskbar to the pinned file icon.

VipraHub

Search Models

mnavena@gmail.com

Dashboard

MODELS

Search Models

My Models

<< Back

## Experiments for model10

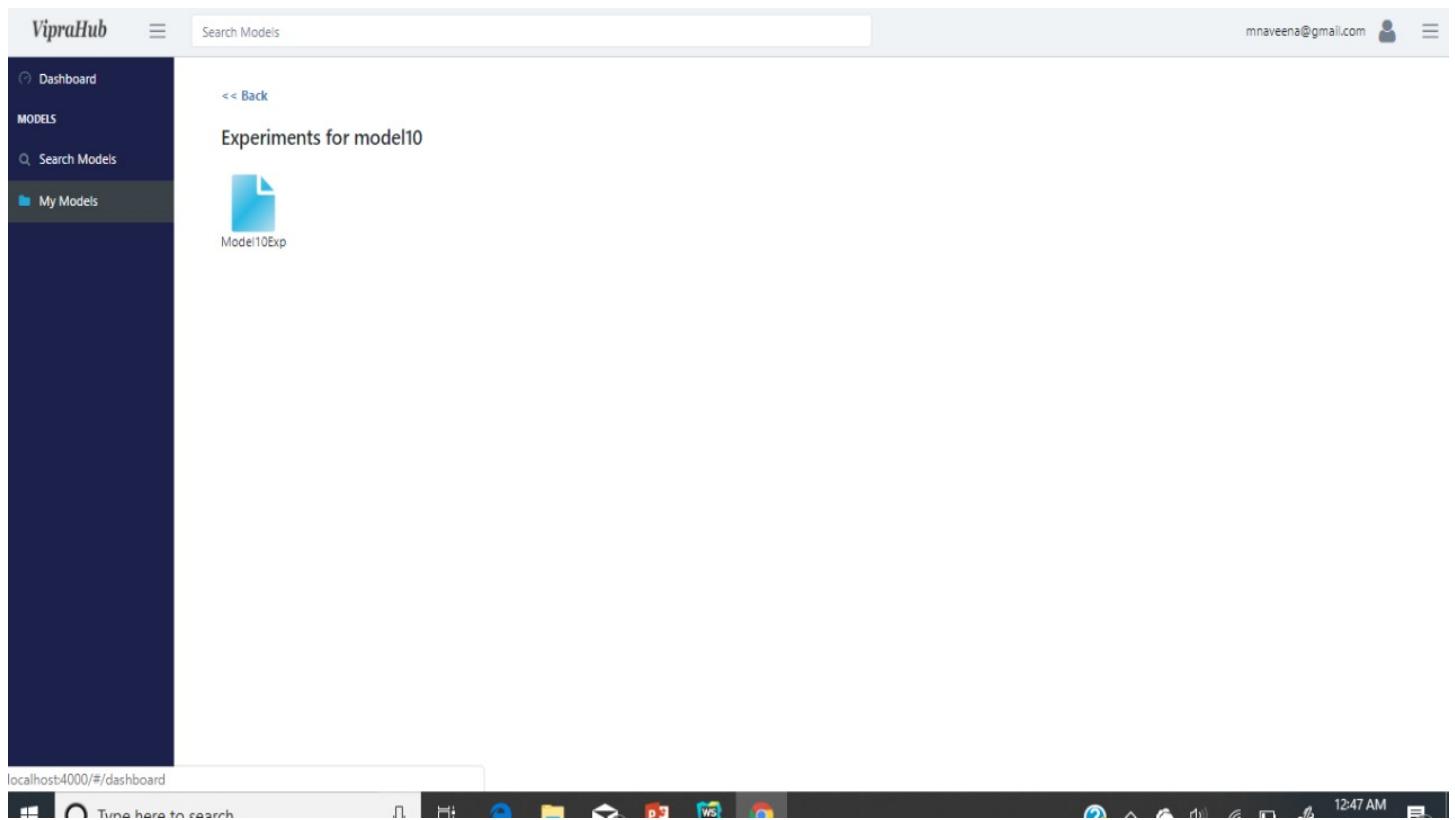


Model10Exp

localhost:4000/#/dashboard

Turn here to search

12:47 AM



Viprahub

Data | Atlas: MongoDB Atlas

https://viprahub.herokuapp.com/#/theme/colors

VipraHub

Fair

dharani@gmail.com

FairnesMdel ★★★★★

Fairness Exp

Download

Rate

OVERVIEW PUBLICATION ARCHITECTURE TEST COMMENTS

DharaniMuli(dharani@gmail.com) posted comment on 2019-05-07T03:28:53.392Z

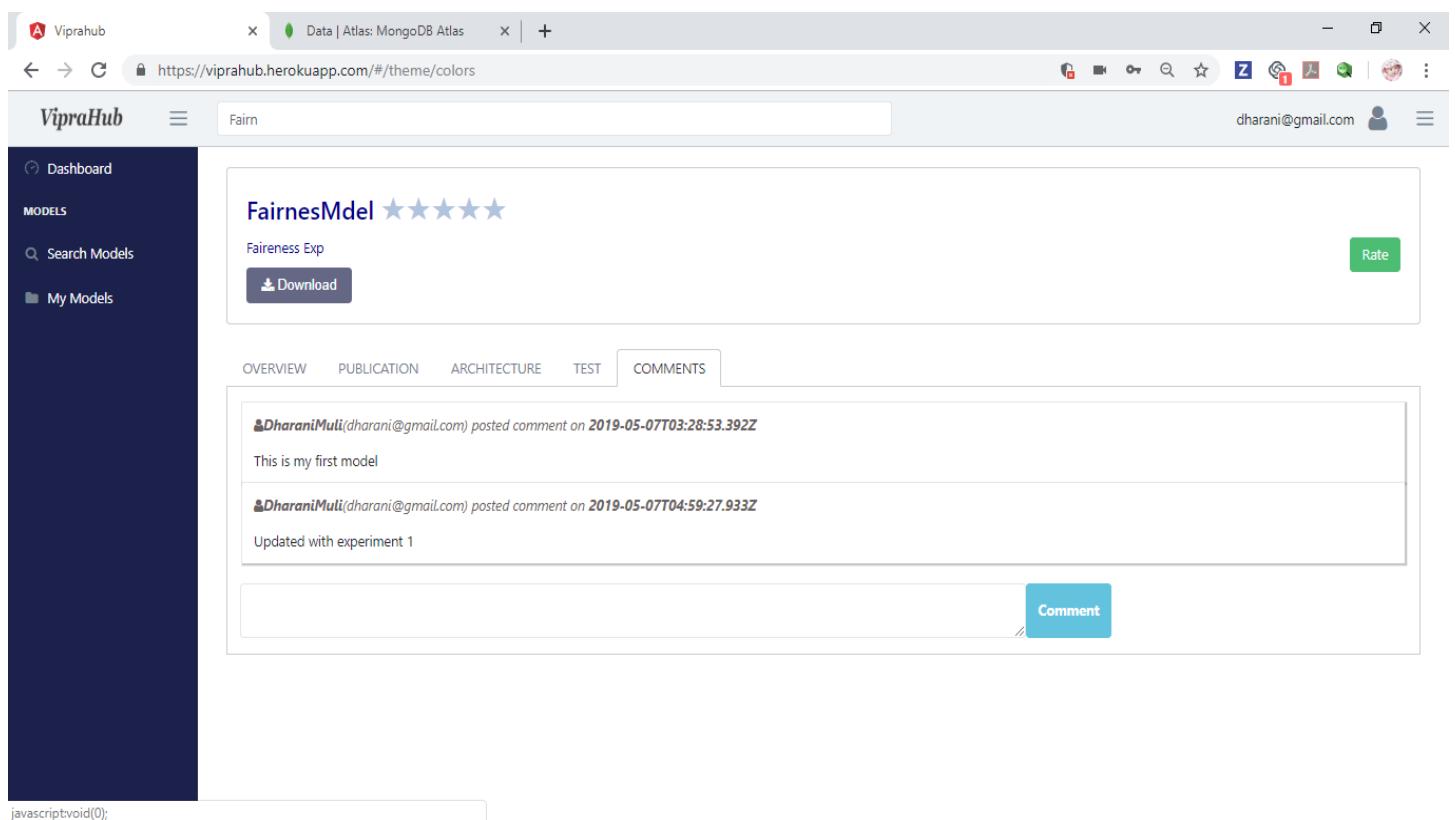
This is my first model

DharaniMuli(dharani@gmail.com) posted comment on 2019-05-07T04:59:27.933Z

Updated with experiment 1

Comment

javascript:void(0);



VipraHub Data | Atlas: MongoDB Atlas https://viprahub.herokuapp.com/#/theme/colors

Fairness Model ★★★★★

Fairness Exp

Download

You can't rate your own model!! Rate

OVERVIEW PUBLICATION ARCHITECTURE TEST COMMENTS

DharaniMuli(dharani@gmail.com) posted comment on 2019-05-07T03:28:53.392Z  
This is my first model

DharaniMuli(dharani@gmail.com) posted comment on 2019-05-07T04:59:27.933Z  
Updated with experiment 1

Comment

VipraHub Data | Atlas: MongoDB Atlas https://viprahub.herokuapp.com/#/theme/colors

Sound Model ★★★★★

SoundExp

Download

Rate

OVERVIEW PUBLICATION ARCHITECTURE

Model Name : Sound Model  
Experiment : SoundExp  
Author : nehanavgale0604@gmail.com  
Category : 5c8df0f71c9d4400002155bd  
Framework : keras 2.2.4  
Size : 1244.224 kilobytes  
Input Tensors : (None, 28, 28)  
Output Tensors : (None, 10)  
Optimizer : Adam  
Loss Function : sparse\_categorical\_crossentropy  
Accuracy Value : 0.974  
Loss Value : 0.072  
Year : N/A

Rate this Model....

★ ★ ★ ★ ★

**Sound Model** ★★★★★

**SoundExp**

**Download**

**OVERVIEW**

**Model Name** : Sound Model  
**Experiment** : SoundExp  
**Author** : nehanavgale0604@gmail.com  
**Category** : 5c8dff0f71c9d4400002155bd  
**Framework** : keras 2.2.4  
**Size** : 1244.224 kilobytes  
**Input Tensors** : (None, 28, 28)  
**Output Tensors** : (None, 10)  
**Optimizer** : Adam  
**Loss Function** : sparse\_categorical\_crossentropy  
**Accuracy Value** : 0.974  
**Loss Value** : 0.072  
**Year** : N/A

## Data Base Design

**viprahub**

DATABASE SIZE: 147.53MB INDEX SIZE: 520KB TOTAL COLLECTIONS: 8

Collection Name	Documents	Documents Size	Documents Avg	Indexes	Index Size	Index Avg
categories	5	262B	53B	1	36KB	36KB
comments	7	1.23KB	180B	1	36KB	36KB
modelsmetadatas	11	5.08KB	474B	2	72KB	36KB
ratings	35	4.8KB	141B	1	36KB	36KB
registeredusers	5	887B	178B	4	144KB	36KB
uploadFiles.chunks	680	147.5MB	222.12KB	2	88KB	44KB
uploadFiles.files	105	19.29KB	189B	2	72KB	36KB
uploads	12	3.38KB	289B	1	36KB	36KB

System Status: All Good Last Login: 134.193.69.153  
 ©2019 MongoDB, Inc. Status Terms Privacy Atlas Blog Contact Sales

Data | Atlas: MongoDB Atlas    +

https://cloud.mongodb.com/v2/5c7f12e2014b7622a6d06c9a#metrics/replicaSet/5c86aaed014b76b3374b8b0b/explorer/viprahub

mongoDB. Atlas All Clusters Eastern Time (US & Canada) Usage This Month:\$0.00 details Dharani

CONTEXT Project 0 DATABASES: 2 COLLECTIONS: 9 REFRESH

PROJECT Clusters Alerts 0 Backup Access Settings Stitch Charts Docs Support

+ Create Database NAMESPACES

viprahub

DATABASE SIZE: 147.53MB INDEX SIZE: 520KB TOTAL COLLECTIONS: 8 CREATE COLLECTION

Collection Name	Documents	Documents Size	Documents Avg	Indexes	Index Size	Index Avg
categories	5	262B	53B	1	36KB	36KB
comments	7	1.23KB	180B	1	36KB	36KB
modelsmetadata	11	5.08KB	474B	2	72KB	36KB
ratings	35	4.8KB	141B	1	36KB	36KB
registeredusers	5	887B	178B	4	144KB	36KB
uploadFiles.chunks	680	147.5MB	222.12KB	2	88KB	44KB
uploadFiles.files	105	19.29KB	189B	2	72KB	36KB
uploads	12	3.38KB	289B	1	36KB	36KB

System Status: All Good Last Login: 134.193.69.153  
©2019 MongoDB, Inc. Status Terms Privacy Atlas Blog Contact Sales

Data | Atlas: MongoDB Atlas    +

https://cloud.mongodb.com/v2/5c7f12e2014b7622a6d06c9a#metrics/replicaSet/5c86aaed014b76b3374b8b0b/explorer/viprahub

mongoDB. Atlas All Clusters Eastern Time (US & Canada) Usage This Month:\$0.00 details Dharani

CONTEXT Project 0 DATABASES: 2 COLLECTIONS: 9 REFRESH

PROJECT Clusters Alerts 0 Backup Access Settings Stitch Charts Docs Support

+ Create Database NAMESPACES

viprahub

DATABASE SIZE: 147.53MB INDEX SIZE: 520KB TOTAL COLLECTIONS: 8 CREATE COLLECTION

Collection Name	Documents	Documents Size	Documents Avg	Indexes	Index Size	Index Avg
categories	5	262B	53B	1	36KB	36KB
comments	7	1.23KB	180B	1	36KB	36KB
modelsmetadata	11	5.08KB	474B	2	72KB	36KB
ratings	35	4.8KB	141B	1	36KB	36KB
registeredusers	5	887B	178B	4	144KB	36KB
uploadFiles.chunks	680	147.5MB	222.12KB	2	88KB	44KB
uploadFiles.files	105	19.29KB	189B	2	72KB	36KB
uploads	12	3.38KB	289B	1	36KB	36KB

System Status: All Good Last Login: 134.193.69.153  
©2019 MongoDB, Inc. Status Terms Privacy Atlas Blog Contact Sales

Data | Atlas: MongoDB Atlas

<https://cloud.mongodb.com/v2/5c7f12e2014b7622a6d06c9a#metrics/replicaSet/5c86aaed014b76b3374b8b0b/explorer/viprahub>

**mongoDB. Atlas** All Clusters

CONTEXT Project 0 DATABASES: 2 COLLECTIONS: 9 REFRESH

PROJECT Clusters Alerts 0 Backup Access Settings Stitch Charts Docs Support

+ Create Database NAMESPACES viprahub

hackthon viprahub categories comments modelsmetadatas ratings registeredusers uploadFiles.chunks uploadFiles.files uploads

**viprahub**

DATABASE SIZE: 147.53MB INDEX SIZE: 520KB TOTAL COLLECTIONS: 8 CREATE COLLECTION

Collection Name	Documents	Documents Size	Documents Avg	Indexes	Index Size	Index Avg
categories	5	262B	53B	1	36KB	36KB
comments	7	1.23KB	180B	1	36KB	36KB
modelsmetadatas	11	5.08KB	474B	2	72KB	36KB
ratings	35	4.8KB	141B	1	36KB	36KB
registeredusers	5	887B	178B	4	144KB	36KB
uploadFiles.chunks	680	147.5MB	222.12KB	2	88KB	44KB
uploadFiles.files	105	19.29KB	189B	2	72KB	36KB
uploads	12	3.38KB	289B	1	36KB	36KB

System Status: All Good Last Login: 134.193.69.153  
©2019 MongoDB, Inc. Status Terms Privacy Atlas Blog Contact Sales

Data | Atlas: MongoDB Atlas

<https://cloud.mongodb.com/v2/5c7f12e2014b7622a6d06c9a#metrics/replicaSet/5c86aaed014b76b3374b8b0b/explorer/viprahub/comments/find>

**mongoDB. Atlas** All Clusters

CONTEXT Project 0 hackthon viprahub

PROJECT Clusters Alerts 0 Backup Access Settings Stitch Charts Docs Support

categories comments modelsmetadatas ratings registeredusers uploadFiles.chunks uploadFiles.files uploads

INSERT DOCUMENT FILTER {"filter": "example"} Find Reset

QUERY RESULTS 1-7 OF 7

```
_id: ObjectId("5cd09c4cb01c3909e426bdd1")
comments: "This is my first model"
modelID: "5cd05056fa992454cf4d6f"
emailID: "dharani@gmail.com"
fullName: "DharaniMuli"
postedDate: 2019-05-06T20:42:52.088+00:00
__v: 0
```

```
_id: ObjectId("5cd0fb754a18390004cb1e7d")
comments: "This is my first model"
modelID: "5cd0c41c3364132e2cf1923b"
emailID: "dharani@gmail.com"
fullName: "DharaniMuli"
postedDate: 2019-05-07T03:28:53.392+00:00
__v: 0
```

```
_id: ObjectId("5cd110af5458c3000406f440")
comments: "Updated with experiment 1"
modelID: "5cd0c41c3364132e2cf1923b"
emailID: "dharani@gmail.com"
```

Type here to search

Data | Atlas: MongoDB Atlas

https://cloud.mongodb.com/v2/5c7f12e2014b7622a6d06c9a#metrics/replicaSet/5c86aaed014b76b3374b8b0b/explorer/viprahub/comments/find

mongoDB. Atlas All Clusters

Eastern Time (US & Canada) Usage This Month:\$0.00 details Dharani

CONTEXT Project 0

PROJECT Clusters Alerts 0 Backup Access Settings Stitch Charts Docs Support

hackthon

viprahub

categories

comments

modelsmetadata

ratings

registeredusers

uploadFiles.chunks

uploadFiles.files

uploads

FILTER {"filter": "example"} INSERT DOCUMENT Find Reset

QUERY RESULTS 1-7 OF 7

`_id: ObjectId("5cd09c4c801c3909e426bdd1")  
comments: "This is my first model"  
modelID: "5cd09c4c801c3909e426bdd1"  
emailID: "dharani@gmail.com"  
fullName: "DharaniMuli"  
postedDate: 2019-05-06T20:42:52.088+00:00  
__v: 0`

`_id: ObjectId("5cd0fb754a18390004cb1e7d")  
comments: "This is my first model"  
modelID: "5cd0c41c3364132e2cf1923b"  
emailID: "dharani@gmail.com"  
fullName: "DharaniMuli"  
postedDate: 2019-05-07T03:28:53.392+00:00  
__v: 0`

`_id: ObjectId("5cd110af5458c3000406f440")  
comments: "Updated with experiment 1"  
modelID: "5cd0c41c3364132e2cf1923b"  
emailID: "dharani@gmail.com"`

Type here to search

15:37 07-05-2019

Data | Atlas: MongoDB Atlas

https://cloud.mongodb.com/v2/5c7f12e2014b7622a6d06c9a#metrics/replicaSet/5c86aaed014b76b3374b8b0b/explorer/viprahub/uploads/find

mongoDB. Atlas All Clusters

Eastern Time (US & Canada) Usage This Month:\$0.00 details Dharani

CONTEXT Project 0

PROJECT Clusters Alerts 0 Backup Access Settings Stitch Charts Docs Support

hackthon

viprahub

fileReferenceIDs

userId

categoryID

name

experiment

metaID

`_id: ObjectId("5cd0c41b3364132e2cf1923a")  
> fileReferenceIDs: Array  
userId: "dharani@gmail.com"  
categoryID: "5cd0f10a1c9d4400002155be"  
name: "Fairness Model"  
experiment: "Fairness Exp"  
metaID: "5cd0c41c3364132e2cf1923b"  
__v: 0`

`_id: ObjectId("5cd0c85c1176904c88f7685d")  
> fileReferenceIDs: Array  
userId: "nehanavagle004@gmail.com"  
categoryID: "5cd0f10a1c9d4400002155be"  
name: "Model4"  
experiment: "Exp2"  
metaID: "5cd0c85d1176904c88f7685e"  
__v: 0`

`_id: ObjectId("5cd0d9f00b15f351006da864")  
> fileReferenceIDs: Array  
userId: "nehanavagle004@gmail.com"  
categoryID: "5cd0f1191c9d4400002155bf"  
name: "Pattern Recognition"  
experiment: "Pattern"  
metaID: "5cd0d9f00b15f351006da865"  
__v: 0`

Type here to search

15:36 07-05-2019



Please also find the presentation in the given below link:

[https://github.com/NehaNavgale/VipraHub/blob/master/Requirements/VipraHub\\_FinalPresentation\\_Team-3.pptx](https://github.com/NehaNavgale/VipraHub/blob/master/Requirements/VipraHub_FinalPresentation_Team-3.pptx).

## 5. PROJECT PROPOSAL AND PLAN

### Project Goals and Objectives

#### Motivation:

There are many model repository sites and hosting sites available but there is no one stop place for the end user to have both the features. In order to overcome this problem, we are motivated to create a single application with high end features like share, test, deploy and visualize the output online.

#### Significance/Uniqueness:

Our application provides following unique features:

1. Model repository with categorization of models.
2. Page to track and visualize the series of experiments.
3. Test and Deploy online.
4. Share and Vote the model.
5. Download the model.

## **Objective:**

Our objective is to design an open source application with maximum features which promote users to do reproducible research, manage all their models at one place. Moreover, this will also provide great visualization and exposure of contributor's work.

## **SCOPE OF THE PROJECT/PROJECT PLAN**

### **Systems Features:**

1. Create user
2. Edit user profile
3. User Dashboard
4. Upload a model
5. Model Profile
6. Download a model
7. Categorization of model
8. Repository Accessibility
9. Vote and comment on model
10. Hosting a model
11. Testing a model
12. Track of experiments

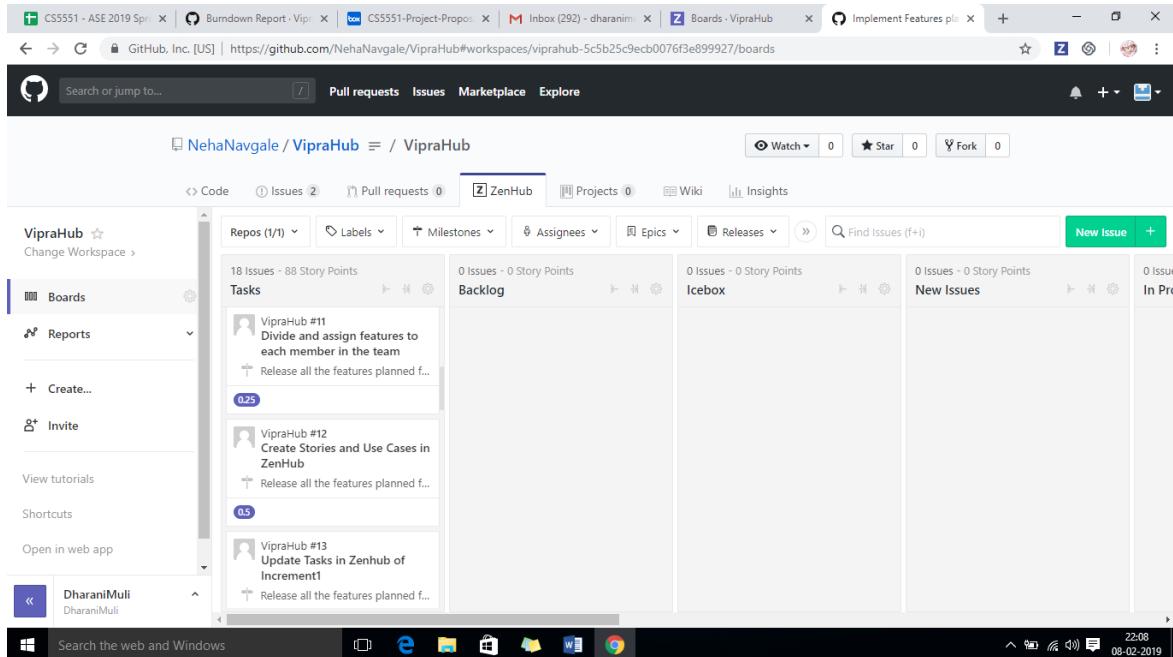
### **Project Timelines: Mile Stones and Tasks**

1. Create Project Proposal (End Date :8th Feb)
  - a. Set up team meeting
  - b. Discuss and close Project proposal and plan
  - c. Create a Zen Hub repository
  - d. Submit plan
2. Wireframes Sign Off from Project Manager (End Date: 18th Feb)
  - a. Requirement Solicitation
  - b. Design Wireframes
  - c. Get Sign off from Project Manager
  - d. Upload wireframes in ZenHub
3. Release 1 (Increment 1) (End Date: 26th Feb)
  - a. Divide and assign features to each member in the team
  - b. Create Stories and Use Cases in ZenHub
  - c. Update Tasks in Zenhub
  - d. Implement Features: Registration Page, Login and User profile
  - e. Test implemented features
4. Release 2 (Increment 2) (End Date: 19th March)
  - a. Update Tasks Use stories and Cases in ZenHub
  - b. Implement Features:
    - i. Provide feature to search, filter or sort the models
    - ii. Categorization of model
  - c. Test implemented features
5. Release 3 (Increment 3) (End Date: 16th April)

- a. Update Tasks Use stories and Cases in ZenHub
  - b. Release2 Features:
    - i. Vote and Comment on Models
    - ii. Download Model
    - iii. Upload Model
    - iv. User Dashboard
    - v. View Model Description and Architecture
    - vi. Provide Repository Accessibility
  - c. Test implemented features
6. Final Release (End Date: 3rd May)
- a. Update Tasks Use stories and Cases in Zen Hub
  - b. Release2 Features:
    - i. Tracking of Experiments
    - ii. Hosting Model
    - iii. Testing Model
  - c. Test implemented features
  - d. Application deployment.

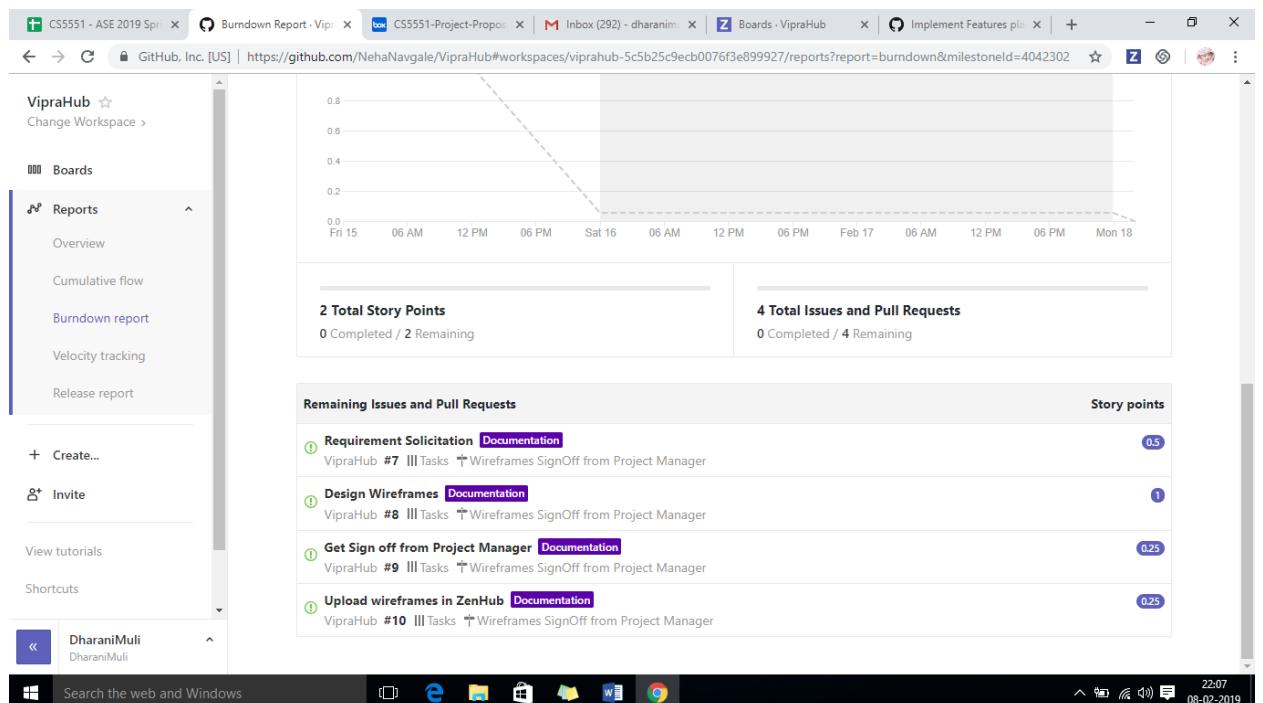
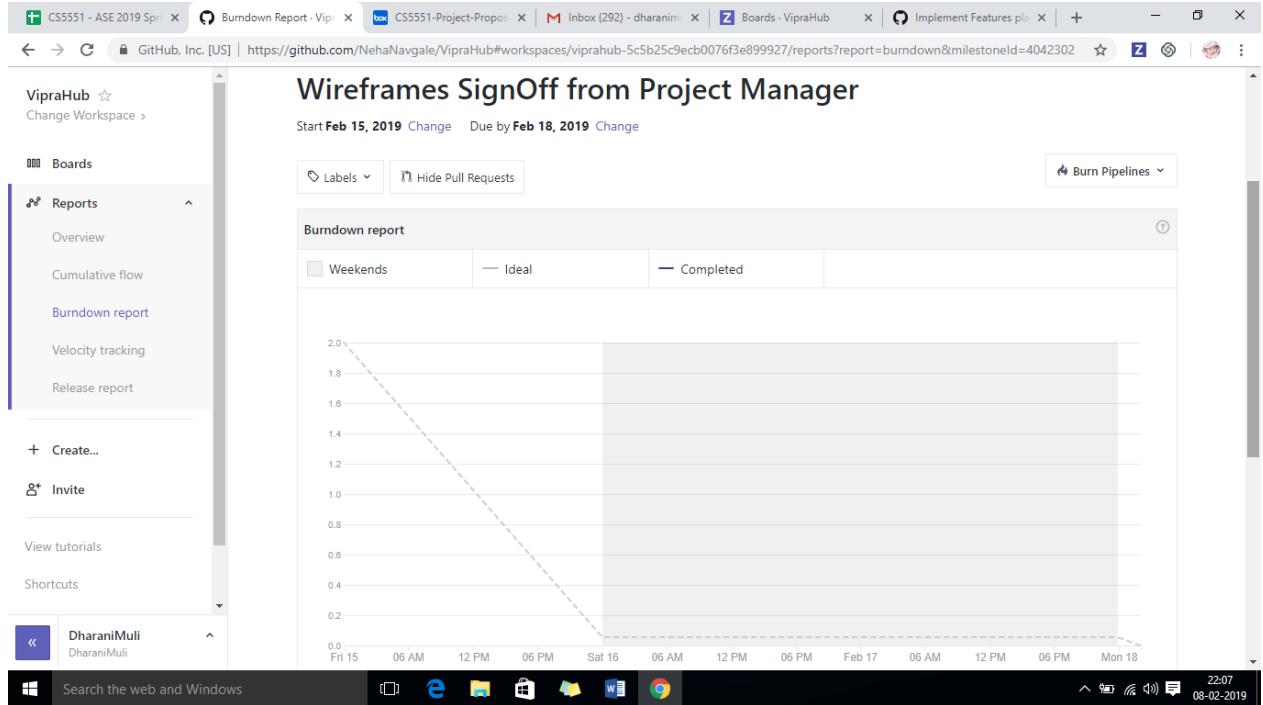
## ZenHub Screenshots

### 1. Created tasks for all increments in ZenHub



### 2. Moved closed tasks to “Closed” Pipeline in Zenhub

3. Created Burndown chart for all the miles stoned, here is the chart for First Milestone “Wireframes Signoff from Project Manager”.



4. Burndown Chart for Closed Tasks

CS5551 - ASE 2019 Spring | Burndown Report - VipraHub | CS5551-Project-Proposal | Inbox (292) - dharanimuli | Boards - VipraHub | Implement Features plan | +

GitHub, Inc. [US] | https://github.com/NehaNavgale/VipraHub#workspaces/viprahub-5c5b25c9eb0076f3e899927/reports?report=burndown&milestoneId=4036255

VipraHub star

Change Workspace >

Boards

Reports

Overview

Cumulative flow

Burndown report

Velocity tracking

Release report

+ Create...

Invite

View tutorials

Shortcuts

DharaniMuli

Search the web and Windows

22:07  
08-02-2019

# Create and Submit Project Proposal & Plan

Start **Feb 4, 2019** Change Due by **Feb 8, 2019 - Due today** Change

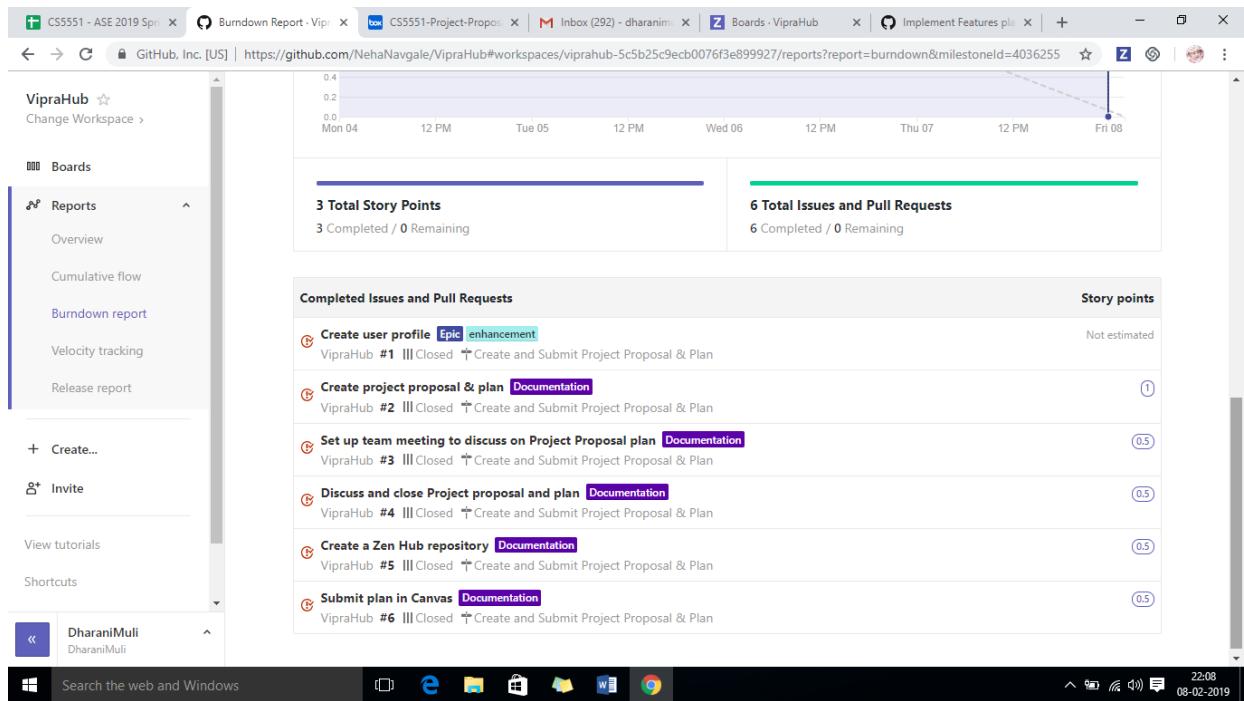
Labels Hide Pull Requests Burn Pipelines

### Burndown report

Weekends Ideal Completed

The burndown chart displays the progress of a project from February 4 to February 8. The Y-axis represents work remaining, ranging from 0.0 to 3.0. The X-axis shows dates: Mon 04, 12 PM, Tue 05, 12 PM, Wed 06, 12 PM, Thu 07, 12 PM, Fri 08. A dashed diagonal line represents the ideal linear progress. A solid blue line represents the actual completed work. The actual progress starts at approximately 2.8 on Monday and decreases to about 0.1 by Friday, indicating significant progress towards the deadline.

Date	Ideal Work Remaining	Actual Completed Work
Mon 04	2.8	~0.1
Tue 05	2.6	~0.1
Wed 06	2.4	~0.1
Thu 07	2.2	~0.1
Fri 08	2.0	~0.1



## TECHNOLOGIES:

1. Angular 7, Typescript
2. CSS3, BootStrap4
3. HTML5
4. Nodejs
5. Mongo DB
6. Heroku

## 6. FIRST INCREMENT REPORT:

## WIREFRAMES AND MOCKUPS

### Home



Search

[Sign In](#) [Sign Up](#) [Contact Us](#)

### Page:

The wireframe shows the homepage layout of the Vipra Hub website. At the top, there is a header with the Vipra Hub logo, a search bar, and navigation links for Sign In, Sign Up, and Contact Us. Below the header, there is a dark blue sidebar containing a brief introduction to the hub, a "Sign Up for VipraHub" button, and a "Browse by Categories" section with buttons for Image Classification, Text and language, Sounds and Music, Fairness and Bias, NLP, Pattern Recognition, and Voice Recognition. The main content area features three rounded rectangular boxes: "Neural Translation with Attention", "CharRNN", and "Text generation using a RNN with...". Each box contains a brief description and a "Browse to see more Models" button at the bottom right. Below these boxes is a section titled "Why Vipra Hub?" with six items: Host your model, Test your model, Visualization, Search other Model, Deploy your model online, and Test your model on VipraHub. At the bottom of the page, there is a footer with copyright information and a note about the management of the hub.

Hub of self-contained deep learning models pretrained for a wide variety of applications. VipraHub highlights recent trends in deep learning applications, enables transfer learning approaches, and promotes reproducible science.

Sign Up for VipraHub

Deep Learning Algorithms

- Deep Neural Networks
- Convolutional Neural Networks
- Recurrent Neural Networks

providing lift for classification and forecasting models  
feature extraction and classification of images  
for sequence of events, language models, time series, etc.

Browse by Categories

Image Classification Text and language Sounds and Music Fairness and Bias NLP

Pattern Recognition Voice Recognition

Neural Translation with Attention

Train a sequence to sequence(seq2seq) model for Spanish to English translation using tf.keras and eager execution.

CharRNN

This notebook will let you input a file containing the text you want your generate to mimic, train your model, see the results, and save it for future use all in one page.

Text generation using a RNN with...

Given a sequence of characters from this data ("Shakespeare"), train a model to predict the next character in the sequence ("e"). Longer sequences of text can be generated by calling the model

Browse to see more Models

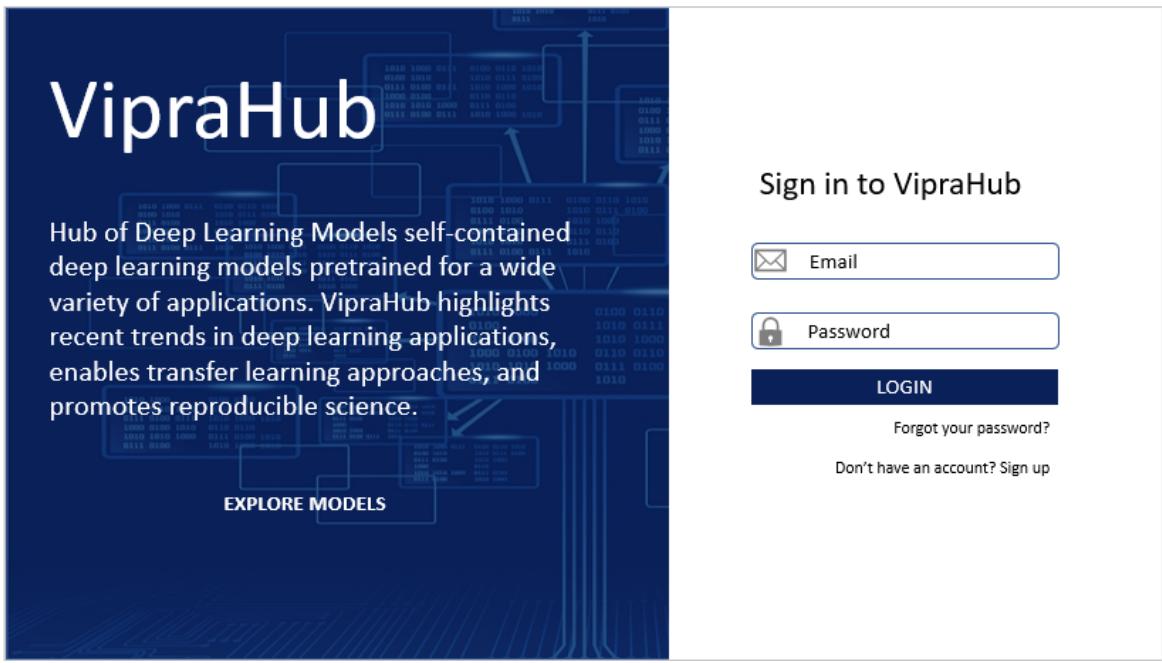
Why Vipra Hub?

- Host your model
- Test your model
- Visualization
- Search other Model
- Deploy your model online
- Test your model on VipraHub and allow others to test and rate your model
- Great Visualization for models and its metadata
- Browse models from a list of different kinds of deep learning categories and sort it based on accuracy and rating

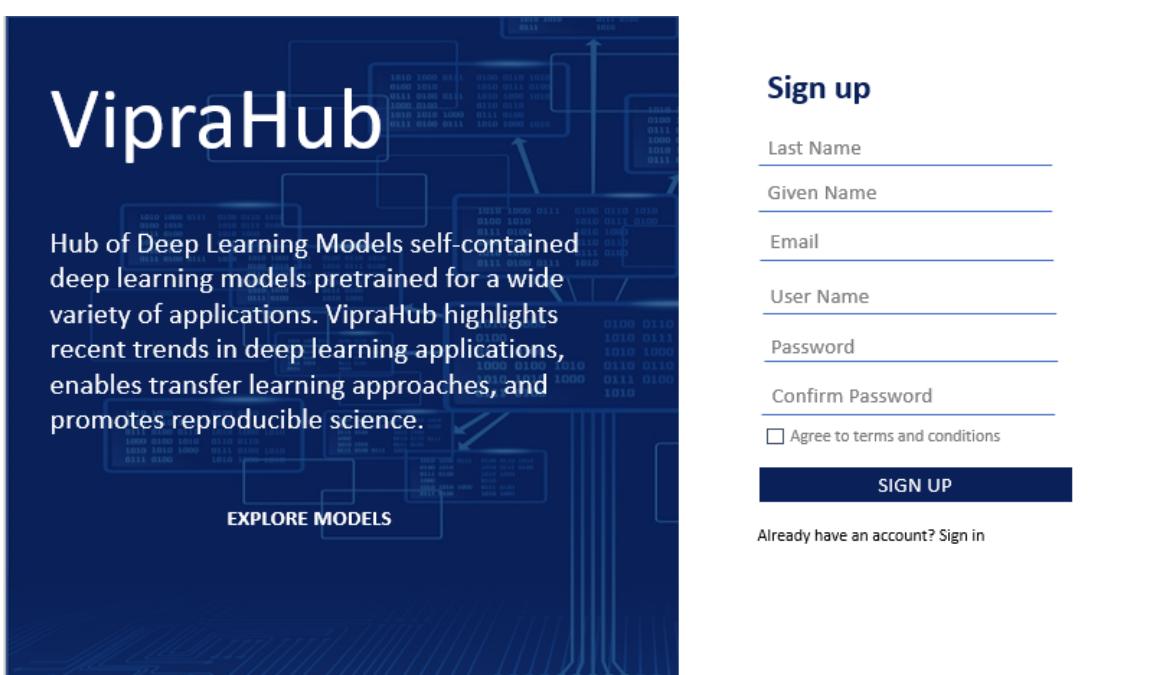
Vipra Hub is managed by developers of University of Missouri Kansas City. Contact us for sharing idea or suggestions

© 2019 Team 3

### Sign in Page:



### Signup Page:



### Search Result Page:

Found 412 matches

**RNN**

Image Classification | Text and language | Sounds and Music | Fairness and Bias | NLP | Pattern Recognition | Voice Recognition

**Neural Translation with Attention**  
Author: James  
Train a sequence to sequence(seq2seq) model for Spanish to English translation using tf.keras and eager execution.  
4.5 stars

**CharRNN**  
Author: Henry  
This notebook will let you input a file containing the text you want your generate to mimic, train your model, see the results, and save it for future use all in one page.  
4.5 stars

**Text generation using a RNN with...**  
Author: Sunanda  
Given a sequence of characters from this data ("Shakespeare"), train a model to predict the next character in the sequence ("e"). Longer sequences of text can be generated by calling the model  
4.5 stars

**Neural Translation with Attention**  
Author: James  
Train a sequence to sequence(seq2seq) model for Spanish to English translation using tf.keras and eager execution.  
4.5 stars

**CharRNN**  
Author: Henry  
This notebook will let you input a file containing the text you want your generate to mimic, train your model, see the results, and save it for future use all in one page.  
4.5 stars

**Text generation using a RNN with...**  
Author: Sunanda  
Given a sequence of characters from this data ("Shakespeare"), train a model to predict the next character in the sequence ("e"). Longer sequences of text can be generated by calling the model  
4.5 stars

## Dashboard:

**Vipra Hub** Search

Dharani Muli dharanimuli559@gmail.com Change password Delete Account Private Public

**Recent Activities**

- Viewed Model1
- Done Experiments on Model2
- Updated profile
- Viewed Comments

No. of Views

Model	No. of Views
MODEL 1	45
MODEL 2	60
MODEL 3	10
MODEL 4	15
MODEL 5	20

**Uploads: 50** **Comments: 55** **Downloads: 30** **Rating: 4.5/5**

Image Classification	Model Name	Ratings (/5)	Downloads
Image Classification	Model 1	3	10
Image Classification	Model 2	4.5	50
Image Classification	Model 3	2	0

## TESTING

- Implemented unit test cases for Login and registration page.
- We have even done manual testing

## IMPLEMENTATION

**Submitted code to GitHub:** <https://github.com/NehaNavgale/VipraHub>

## EXISTING DESIGN/SERVICES/REST API

We are not using any services for Increment 1, hence we have not included any services in this report.

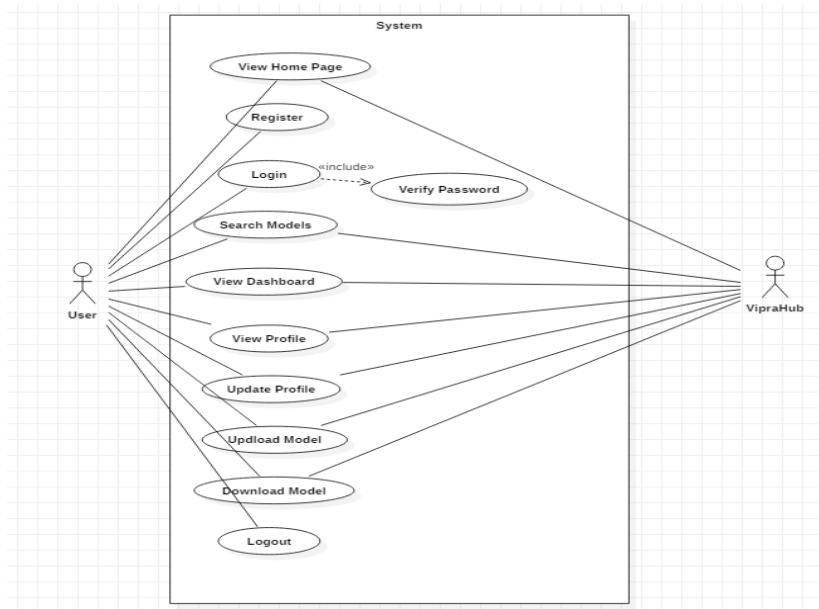
## WRITE USER STORIES /USE CASE/SERVICE DESCRIPTION

We have written stories in Zenhub, kindly click on below link to view

<https://github.com/NehaNavgale/VipraHub/issues/44workspaces/viprahub-5c5b25c9ecb0076f3e899927/boards?repos=168255028>

The screenshot shows the Zenhub interface for the 'VipraHub' workspace. On the left, there's a sidebar with options like Boards, Reports, Create..., Invite, View tutorials, Shortcuts, and Open in web app. The main area has tabs for Code, Issues (30), Pull requests (0), ZenHub (selected), Projects (0), Wiki, and Insights. Below these tabs, there are four columns: 'Tasks' (22 Issues - 87.75 Story Points), 'Backlog' (8 Issues - 4 Story Points), 'Icebox' (0 Issues - 0 Story Points), and 'New Issues' (0 Issues - 0 Story Points). Each column contains several user stories, such as 'VipraHub #12 Create Stories and Use Cases in ZenHub', 'VipraHub #13 Update Tasks in Zenhub of Increment1', and 'VipraHub #14 Implement Features: Registration Page, Login and User profile'. The bottom of the screen shows a Windows taskbar with various icons and the date/time (25-02-2019, 22:02).

## USE CASE DIAGRAMS:



## PROJECT MANAGEMENT:

## **Work Completed:**

### **Description:**

As per the increment 1 plan we have closed Login, Registration, Home and Dashboard (prototype).

### **Responsibilities:**

#### **Neha (Hours16):**

1. Created VipraHub Repository on GitHub.
2. Designed Wireframes for Home page.
3. Implemented Home page.
4. Performed Unit testing for Login and Registration page using Karma and Jasmin.

#### **Naveena (Hours16):**

1. Created Project on WebStorm.
2. Designed wireframes for Model dashboard pages
3. Implemented responsive user dashboard.
4. Used ng2 charts for statistics in user dashboards.
5. Created use case diagrams for user dashboards.

#### **Dharani (Hours16):**

1. Created project structure on ZenHub.
2. Created user stories on Zenhub.
3. Created wireframes for Dashboard.
4. Implemented Login page.

#### **Pavan Kumar (Hours16):**

1. Created wireframes for Login and Registration pages.
2. Implemented Registration page.
3. Bug Fixes for all the pages.
4. Created Use Case diagrams.

## **Work to be Completed:**

1. Retrospective meeting
2. Bug Fixing
3. Update Tasks Use stories and Cases in ZenHub
4. Implement Features:
  - Provide feature to search, filter or sort the models
  - Categorization of model

## **Issues/Concerns:**

None

## **7. SECOND INCREMENT REPORT**

### **SUMMARY OF PROJECT INCREMENT 2**

---

In this increment we focused on upload and search models functionality. User can upload model files on Viprahub in a category by selecting existing categories. User will be able to then view the top models on the home page and can filter based on categories. We have also provided functionality to search models using keyword and sort the search results.

### **DETAIL DESIGN OF FEATURES (USING TOOLS)**

#### **Features 1**

---

- Feature Name: Upload model files on viprahub and store it on mongoDB.
- Feature Description:
  1. User is provided with an upload button.
  2. Once user clicks the upload button popup for uploading model files opens.
  3. In this popup single file upload and multiple file upload features are implemented.
  4. User can select a category.
  5. Uploaded files are saved in mongodb against the selected category.

#### **Features 2**

---

- Feature Name: Extracting metadata from uploaded files on viprahub
- Feature Description: Should read data from the meta file and store the data in the MongoDB so that we can performing searching and sorting on models.

#### **Features 3**

---

- Feature Name: Feature to search models on search page
- Feature Description: User is provided with search option on header. Once user type any keyword and click on search button the search result page will display results based on searched text.

#### **Features 4**

---

- Feature Name: Feature to sort models on search page

- Feature Description: Should be able to sort the search result based on accuracy and year

## Features 5

---

- Feature Name: Feature to filter models on home page based on category
- Feature Description: All the existing categories will be displayed on home page. While clicking on any category name the top 3 models of the clicked category based on accuracy will be displayed.

Submitted code to GitHub: <https://github.com/NehaNavgale/VipraHub>

## EXISTING DESIGN/SERVICES/REST API

---

NodeJS API, mongoDB routing

## PROJECT MANAGEMENT:

### Work Completed:

---

We have completed all the above mentioned 5 features

### Responsibilities:

---

Neha (story points: 5):

1. Implemented functionality to display deep learning categories on Home page
2. Implemented feature to filter models on home page by clicking category name and displaying top 3 models of the selected category
3. Implemented search functionality on search page.
4. Contributed in the wiki documentation

Dharani (story points: 4):

1. Implemented Login and Registration connectivity with MongoDB.
2. Created common services for components communication.
3. Implemented extracting metadata from uploaded files on viprahub.
4. Contributed in the wiki documentation and report submission.

Pavan Kumar (story points: 4):

1. Implemented sorting functionality.
2. Contributed in the wiki documentation and report submission.

Naveena (story points:10):

1. Created Mongo DB atlas cluster
2. Added mongo DB connection and routing setup in server file.
3. Created UI for multiple file upload popup.
4. Implemented multiple file upload functionality using GridFS and multer in mongoDB.
5. Documentation

### **Work to be Completed:**

1. Download a model
2. Categorization of model
3. Repository Accessibility
4. Vote and comment on model
5. Hosting a model
6. Testing a model
7. Track of experiments

### **Issues/Concerns:**

1. Facing issues with reloading one component on the button click on another component.
2. Full text search in mongoDB for all the columns in a collection. Found work around by using \$or aggregation

## **8. THIRD INCREMENT REPORT**

### **SUMMARY OF PROJECT INCREMENT 3**

---

In this increment we have incorporated suggestions received from Professor as a part of Increment 2 along with developing new features. We have also customized the UI of the application to provide user interactive interface. We are also providing below given new features:

1. Feature to download models
2. Advance searching on models
3. View complete details of models in a model dashboard.

### **DETAIL DESIGN OF FEATURES (USING TOOLS)**

---

#### **Features 1**

---

- Feature Name: Download model from MongoDB

- Feature Description:
  - i. User is provided with an download button in user model view page
  - ii. Once user clicks the download button a zipped folder will be downloaded to user's machine.

## Features 2

---

- Feature Name: Advanced Searching of Models
- Feature Description: Advance searching is the extension of search functionality which is implemented as a part of Increment 2.
  - i. In this we have provided feature to filter the search results based on category and rating.
  - ii. Provided different views for search results.

## Features 3

---

- Feature Name: View Page for Model
- Feature Description: View page for model is displayed when user clicks on any model from search result. This view displays the information about Model. The information includes meta data of the model and the architecture.

## Features 4

---

- Feature Name: Models menu
- Feature Description: Menu for displaying all the models names and experiment names.

## Features 5

---

- Feature Name: Carousel on Home page.
- Feature Description: We have provided carousel on home page to market the application.

## IMPLEMENTATION

---

Technologies used: Angular, NodeJS, MongoDB, ExpressJS

---

Submitted code to GitHub: <https://github.com/NehaNavgale/VipraHub>

## EXISTING DESIGN/SERVICES/REST API

---

NodeJS API, mongoDB routing

## **PROJECT MANAGEMENT:**

### **Work Completed:**

---

We have completed all the above mentioned 5 features

### **Responsibilities:**

---

**Neha Navgale**(story points: 10):

1. Advance searching.
2. Completed home page view.

**Dharani Muli** (story points: 10):

1. View page for Model
2. Contributed for Layout customization.

**Pavan Kumar** (story points: 10):

1. Download functionality
2. Customizing the fixed layout

**Naveena Madepally** (story points:10):

1. Modified upload functionality.Added option to upload experiments against models.
2. Added models menu.

### **Work to be Completed:**

1. Vote and comment on model
2. Integration with GitLab
3. Testing of application
4. Bug Fixes and Code Refactoring
5. Deployment of application

## **9. FOURTH INCREMENT**

### **SUMMARY OF PROJECT INCREMENT 4**

---

In this increment we have incorporated suggestions received from Professor as a part of Increment 2 and 3 along with developing new features. Below are the features that was implemented as part of Increment 4:

1. Download uploaded models from MongoDB.
2. Meta Data extraction from MongoDB.
3. Experiment level comments.
4. Made part of Dashboard dynamic.
5. Rate Experiment.
6. Search/filter with Ratings
7. Enhancements in search result, view model and home page.
8. Deployed application into Heroku cloud.

## DETAIL DESIGN OF FEATURES (USING TOOLS)

---

### Features 1

---

- Feature Name: Download model from MongoDB
- Feature Description:
  - i. User is provided with an download button in user model view page
  - ii. Once user clicks the download button, a zipped folder will be downloaded to user's machine.

### Features 2

---

- Feature Name: Meta Data extraction from MongoDB
- Feature Description: Meta data contains the information about the model. As soon as user uploads the model into MongoDB, We are extracting the data from the metadata file and storing it into another table. Now when user navigate to the view experiment page , they will be able to view the meta data information.

### Features 3

---

- Feature Name: Experiment level comments
- Feature Description: User can provide comments for any experiment and also can view the comments posted by other users as well for any experiment.

### Features 4

---

- Feature Name: Made part of Dashboard dynamic
- Feature Description:
  1. User will be able to view the Statistics of total number of models, uploaded, downloaded, comments and ratings
  2. Also will be able to view the top 5 models of his on the dashboard.

## **Features 5**

---

- Feature Name: Rate Experiment
  - Feature Description:
1. User will be able to view the overall rating of any experiment.
  2. User will be able to rate the experiment as well.

## **Features 6**

---

- Feature Name: Search/filter with Ratings
- Feature Description: Now user will be able to search or filter with the rating of the experiments.

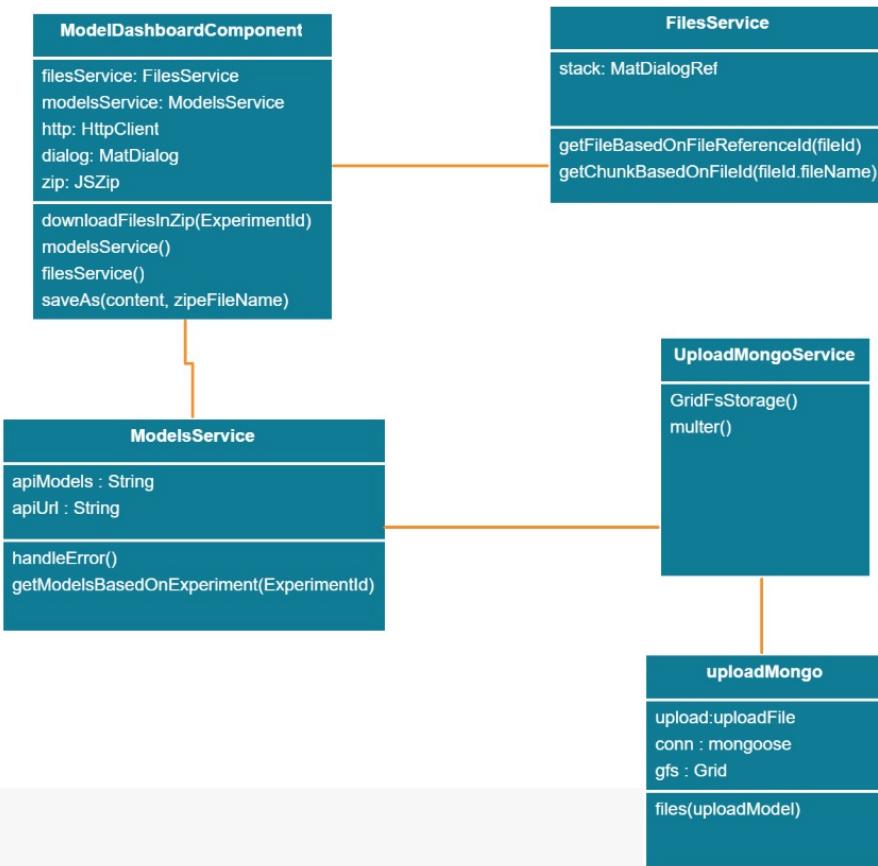
## **Features 7**

---

- Feature Name: Deployment in Heroku
  - Feature Description:
1. We have separated both Front end and Back end into Heroku

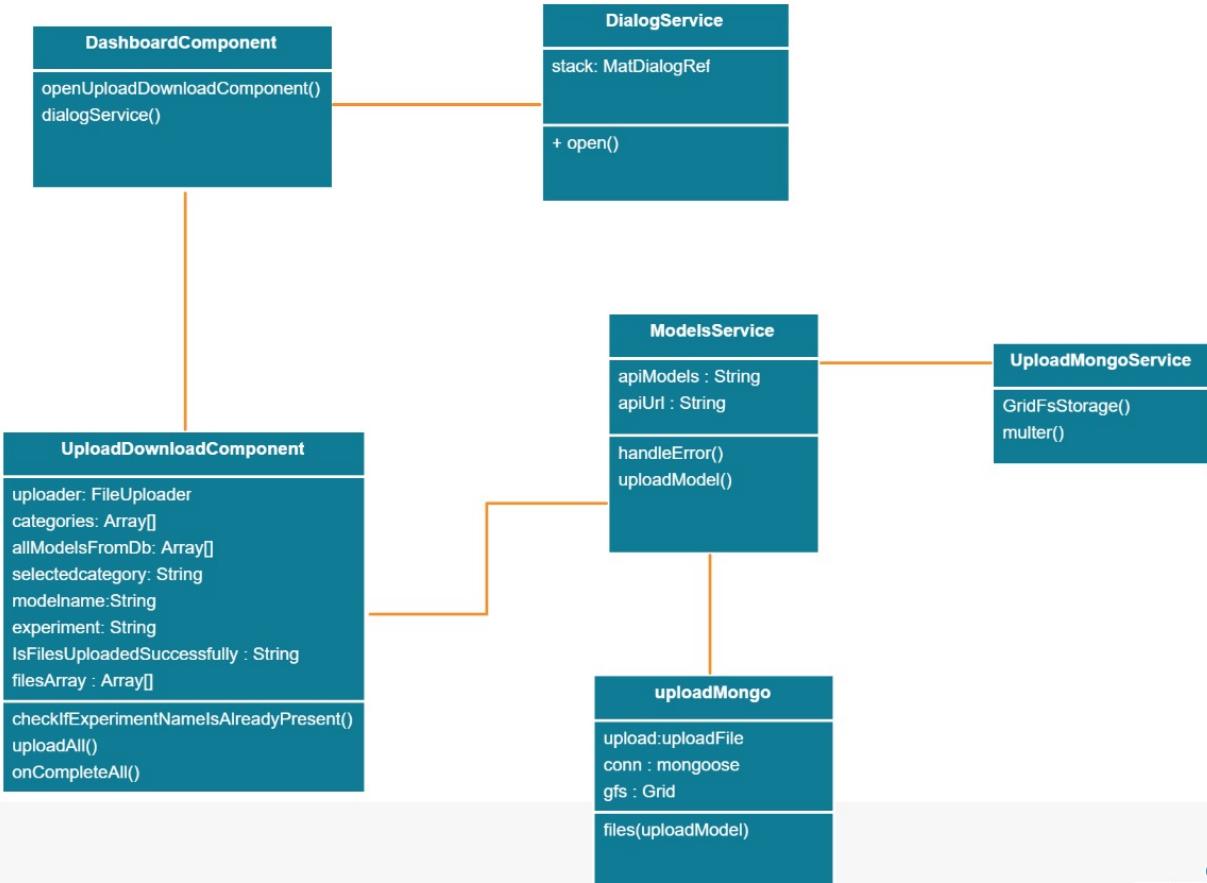
## **UML Diagram for Upload experiment**

### Class Diagram for File Download From Mongodb GridFs



UML Diagram for download experiment

## Class Diagram for File Upload to Mongodb GridFs



creately  
www.createley.com • Online Diagramming

## ZenHub Report

**VipraHub**

**Board**

**Reports**

- Overview
- Cumulative flow New
- Control chart New
- Burndown report
- Velocity tracking
- Release report

**Create...**

**Edit Workspace**

**Invite**

**View tutorials**

**Shortcuts**

**75 Total Story Points**  
75 Completed / 0 Remaining

**8 Total Issues and Pull Requests**  
8 Completed / 0 Remaining

Completed Issues and Pull Requests		Story points
④ Update Tasks, Use stories and Cases in ZenHub for Final Release	VipraHub #22	1
④ Implement Features planned for Final Release	VipraHub #23	17
④ Test implemented features for Final Release and Go live	VipraHub #24	2
④ Implement Features planned for Final Release	VipraHub #25	17
④ Increment -4 Tasks <span>enhancement</span>	VipraHub #49	10
④ Contribution for Final Project Submission <span>Documentation</span>	VipraHub #50	1
④ Increment -4 tasks <span>enhancement</span>	VipraHub #54	10
④ Increment 4 Task 66 <span>enhancement</span>	VipraHub #66	17

## **PROJECT MANAGEMENT:**

### **Work Completed:**

---

We have completed all the above mentioned 7 features and also successfully deployed our application into Heroku. Here is the deployed URL:

<https://viprahub.herokuapp.com/>

### **Responsibilities:**

---

**Neha Navgale**(story points: 10):

1. Made Enhancements in the search result.
2. Made Dashboard dynamic.
3. Deployed backend application into Heroku.
4. Bug fixes.

**Dharani Muli** (story points: 10):

1. Implemented Experiment level comments.
2. Implemented Experiment level Ratings.
3. Made Enhancements in the view model.
4. Bugfixes.

**Pavan Kumar** (story points: 10):

1. Extracted Meta data from MongoDB and integrated with the view model
2. Deployed front end application
3. BugFixes.

**Naveena Madepally** (story points:10):

1. Implemented Download from MongoDB Feature
2. BugFixes.

## **10. REFERENCES:**

Model Hub: <http://app.modelhub.ai>

Google Seedbank: <https://research.google.com/seedbank/>

GitHub

## **11. ACKNOWLEDGEMENT STATEMENTS:**

"The work has been completed under the guidance of Dr. Yugi Lee, Rajaram Anantharaman, and TAs (Sirisha Rella, Bhargavi Nadendla ) in CS5551 Advanced Software Engineering, University of Missouri-Kansas City), Spring 2019.