

Online Host, Visualization, and Deployment

**Submitted by**

**Vamsi Draksharam: Class id -9**

**Goutham Gandreddi: Class Id -11**

**Premchand Lingamgunta: Class id - 15**

**Project proposal:**

**Motivation:**

With the increasing number of models and iterative tests on the models, the need for a system to capture and host a model, visualize the results and sharing the trained models online becomes a critical component of the overall deep learning lifecycle.

**Significance / Uniqueness:**

* It’s a community/platform to host the user models and allow others to test it, understand as well as download the code to local.
* The user can view all his models at one stop.

**Objective:**

As mentioned above, ‘online host, visualization and deployement’ gives the user a platform where user can upload or download the models or pretty much anything. User can view any model as a separate module and can comment and rate the model.

**System Features:**

* The Application is planned to build with a MEAN stack approach using Angular 6 version and will be putting efforts to work towards creating an IONIC framework application as-well.
* The Discussion form for each issue is going to have a thread of comments and other features which are planning to be implemented

**Project Plan:**

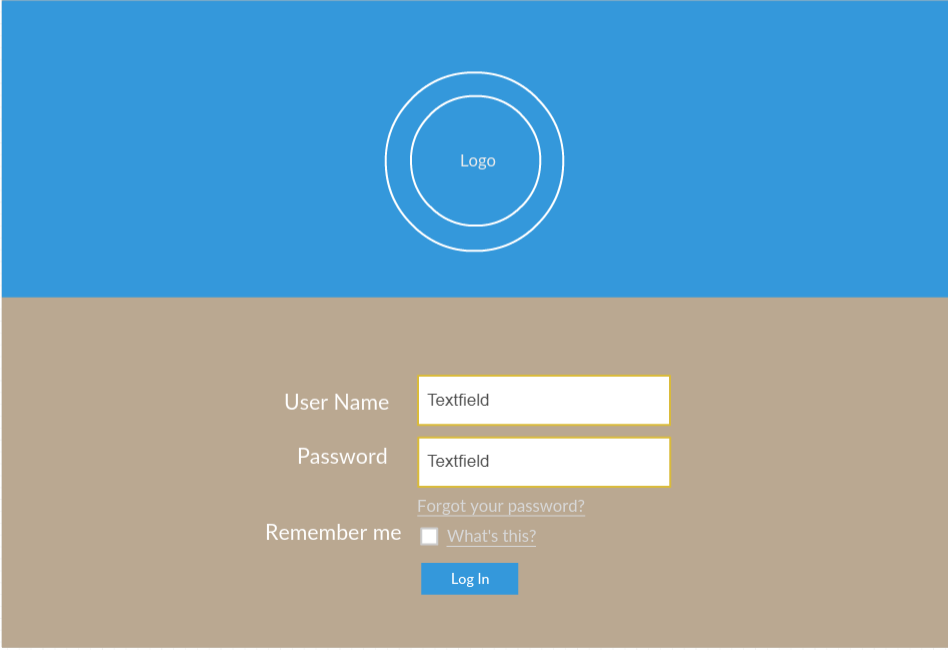
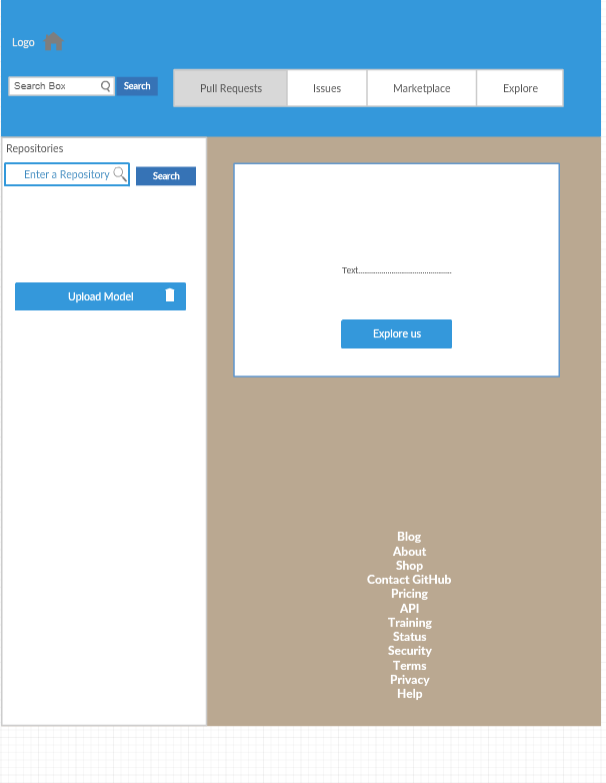
**Schedule for Four Increments:**

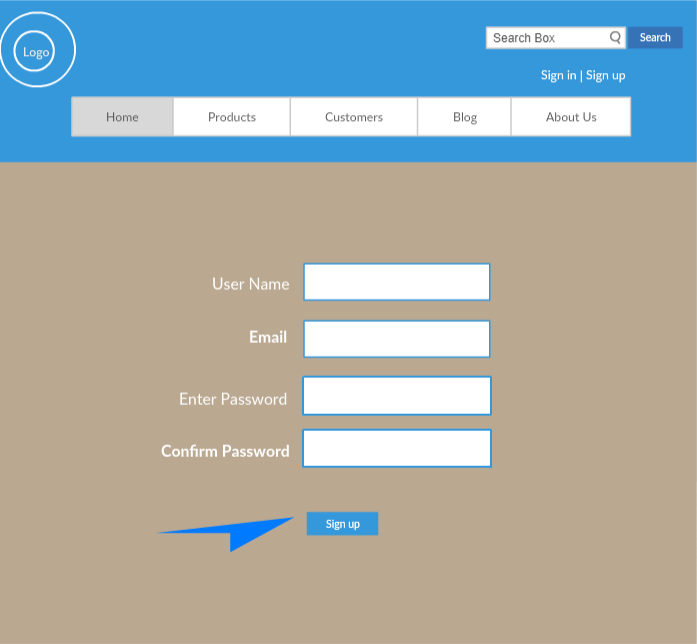
**Increment 1 – Create Sign-in , sign-up pages & Home Page (navigation bar/menu bar)**

The increment is mainly divided into three sections,

* Sign-up: Creating the signup page and authenticating all the fields based on user input on various fields of user data.
* Sign-in: Creating the sign-in page and authentication if he is a registered user or not.
* Home Page: The Home page has the menu for navigating through features like, displaying the menu for user information, user models, search bar for searching the models and sign-out features.
* Create the required tables for the user authentication and registration details for validation purposes and test the data.

**Below is the wire frame diagram for the first increment pages: -**





**Increment 2 – Create User Profile, Menu bar & Implement search Feature**

Creating the User Profile – Provides the user information and his available models and navigate to the specific models on clicking them.

Menu Bar – Adding the implementation for user profile, user models, logout features and redirect and create their respective views.

Implement Search Feature -

1. Simple search feature – is with either repository name/ user name.
2. Sorting the models - The power to filter is handed over to the user, resulting in getting more refined results, as user desires and sort the results by best match, most likes, recently implemented, etc.

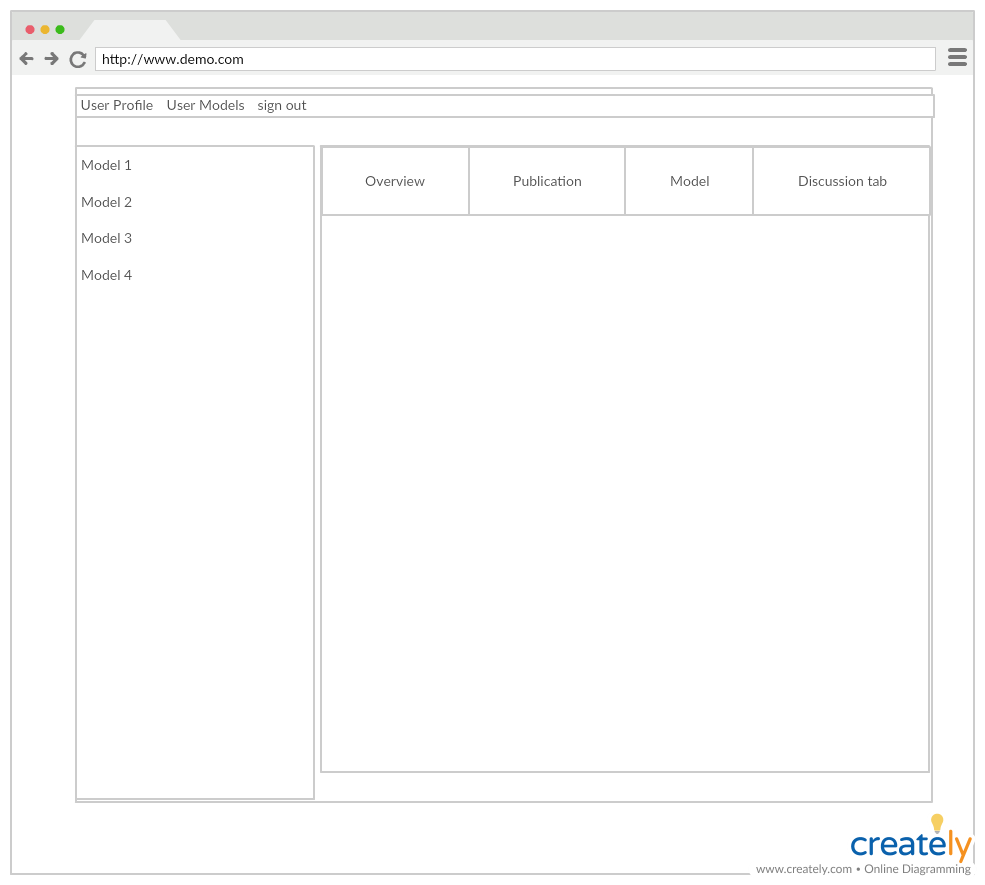
**Increment 3 – Implementation for Model view:**

Upon clicking the desired model, the tab view containing the various tabs like overview, publication, model and discussion.

Features like-

* Overview – contains the details about meta data like description, Model name, etc.
* Publication – contains the details of title for the model, author details, abstract, publication, etc.
* Model – displays the models and its various nodes and their features.
* Discussion tab – it will be implemented in increment 4.

**Below is the wire frame diagram for Model View: -**



**Increment 4 – Issue Discussion form cont. & testing:**

The Idea is - like a Facebook post - to give multiple users to upvote, comment, involve in providing solutions to certain issues.

Challenges:

* Handling the nested comments section.
* How to store and retrieving the comments from the Database.
* How to display the comments in the UI.

The implementation of the discussion tab features like giving the proper view as it grows in size.

The end to end testing of the application with all its features needs to be done and need to keep some time for any fixes after integrating all the modules of the application.