Project Report

Title: eDeals (all the best deals in one place!)

Authors:

Bharath Krishnan (bharath@ccs.neu.edu)
Rifat Hasan (rifat@ccs.neu.edu)
Umang Mehta (umang12@ccs.neu.edu)

Abstract:

This report helps in understanding the basic purpose and functioning of the eDeals website. It describes the different actors involved, their roles and the way they interact with the website.

Introduction:

The eDeals site simplifies your online shopping experience. For any product that you may wish to purchase online, the eDeals website displays options from various online dealers and also recommends the best choice to the user.

Requirements:

Use Case Description Ontology:

System-Actors

User [User]

Description:

- Searches for deals for the product required.
- Is presented with a list of deals for the product searched. User can filter products based on requirements.
- User chooses the preferred deal and proceeds to the corresponding site.
- Can refer deals to other users.
- Receives discounts if selecting deals while logged in.

Kind Of: User

Admin: [Admin]

Description:

- Admin will provide the interface to the user for searching best deals.
- Admin will validate user account upon registering.
- Admin will create and provide ads on the eDeal site

Kind Of: Admin

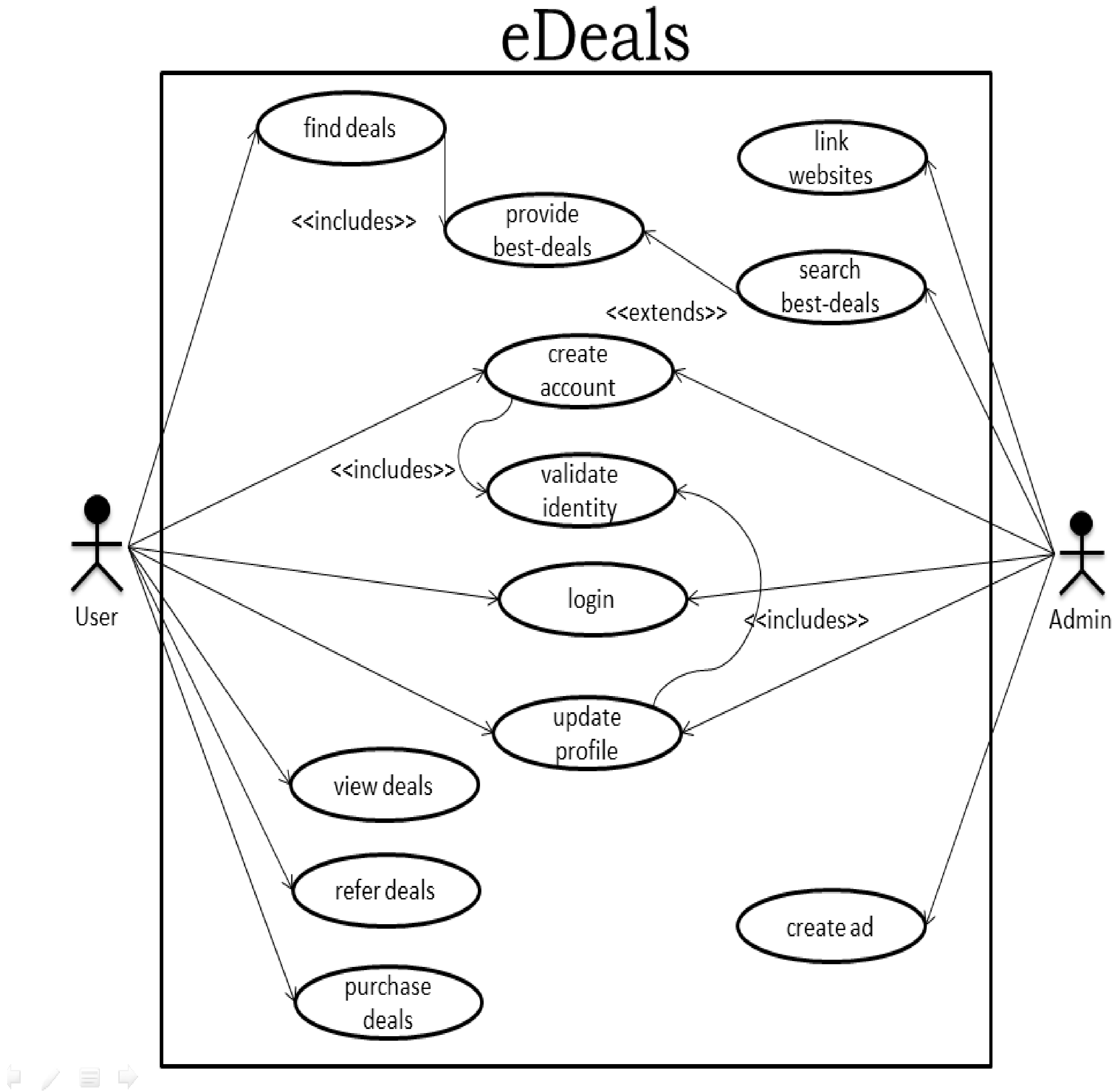


Figure: Use Case Analysis of eDeals

Use Case: Find Deals

Find a deal: [findDeals]

- Description: User will enter product name and search for the best deal based on his requirement and system availability.
- Precondition: User should be on eDeals website/application.
- Step by step:
  - User enters the particular product which he requires in search box.
  - System provides the deals that are relevant to the product
  - System recommends the best deal.
  - User can filter the deals based product attributes.

  - System provides the deal based on user requirement.

Use Case: Create Account

Create an account: [createAccount]

- Description: User creates an Account to find the Best deal by entering his first name, last name, date of birth, telephone number, shipping address and email address.
- Precondition: User should have opened the eDeals site.
- Step by step:
  - User opens the eDeals site
  - User keys in his first name, last name, date of birth, telephone number and email address to create an account
  - Admin sends an email to the user.
  - User clicks on the link in the email.
  - System redirects the user to the eDeals website and logs the user into the system and activates the account.
  - User clicks on the link provided by the system and activates the account.

Use Case: Validate Identity

Validate users Identity: [validateIdentity]

- Description: The admin validates the details entered by the user while creating an account.
- Precondition:
  - User should have created an account.
  - User should be a human being.
- Step by step:
  - Admin logs in to the system.
  - Admin authenticates the user's inputs.
  - System automatically sends a mail to the user with a link to activate account upon admin's notification.

Use Case: Login

User Login to the system: [login]

- Description: The User Login/Sign-in into the system after the admin validates his Identity.
- Precondition:
  - User should have created an account.
  - User should have a valid password.
  - User should be validated by the Admin.
- Step by step:
  - User enters his User-ID and valid Password.
  - Admin provides an captcha to check whether the user is a Human-Being or a computer generated program
  - System automatically allows the User to enter his Homepage on Admin's authentication of valid User.

Use Case: Update Profile

Updating the User's Profile: [updateProfile]

- Description: The user can update the profile details which include last name, first name, birth date, telephone number, shipping address and email address. The user can also add a profile picture to the profile.
- Precondition:
  - User should have created an account.
  - User should have valid password to login
  - User account should be validated by admin
- Step by step:
  - User logs in to the account.
  - User changes personal information.
  - User may add a photo.

Use Case: View Deals

View a deal: [viewDeals]

- Description: User will view the deals with details provided by the system.
- Precondition:
  - System should provide the deals based on user requirement.
  - User should have an account and must be logged into the system.
- Step by step:
  - User will view the deals based on his requirement.
  - User can choose any of the provided deals to view in detail given by the system
  - System will provide all the details relevant to that particular deal along with the offer that the system provides on the original deal.

Use Case: Refer Deals

Refer a deal: [referDeals]

- Description: User can refer the deals to anyone.
- Precondition: User must be logged into the system.
- Step by step:
  - User logs into the system.
  - Admin authenticates the user.
  - User refers the viewed deal to anyone.

Use Case: Purchase Deals

Purchase a deal: [purchaseDeals]

- Description: User purchase a deal given by the system.
- Precondition: User must be logged into the system.
- Step by step:
  - User logs into the system.
  - Admin authenticates the user.
  - User selects a deal to purchase.
  - System redirects the user to the parent deal site for payment option

Use Case: Link Websites

Providing links to the websites: [linkWebsites]

- Description: The system finds the deals according to the user's requirements and provides the user with a link to view the deals.
- Precondition:
    - User should have opened the eDeals website
    - User should have searched for a product
- Product should be a valid one.
- Step by step:
    - User searches for a product.
    - System provides various deals that would redirect to different websites.

Use Case: Search best-deals

Search the best deals: [searchBestdeals]

- Description: The system will search the best deals for the products.
- Precondition: The system must have linked the web services to search best deals

Step by step:

- System links the web services.
- System searches the best deals from the linked web services.

Post Condition: The search should be filtered by the users search criteria

Use Case: Provide best-deals

Provide the best deals: [provideBestdeal]

- Description: The system will provide the best deals for the product based on user requirement and also provide a link where the user can have discount using eDeals site.
- Precondition: User should have provided the searching criteria by finding deals
- Step by step:
    - System gets the user search criteria when user tries to find deals.
    - System links that search with web services.
    - System gets data from the web and provides it in the eDeal site.

Use Case: Create Ads

Admin Creates Ad on the system: [createAd]

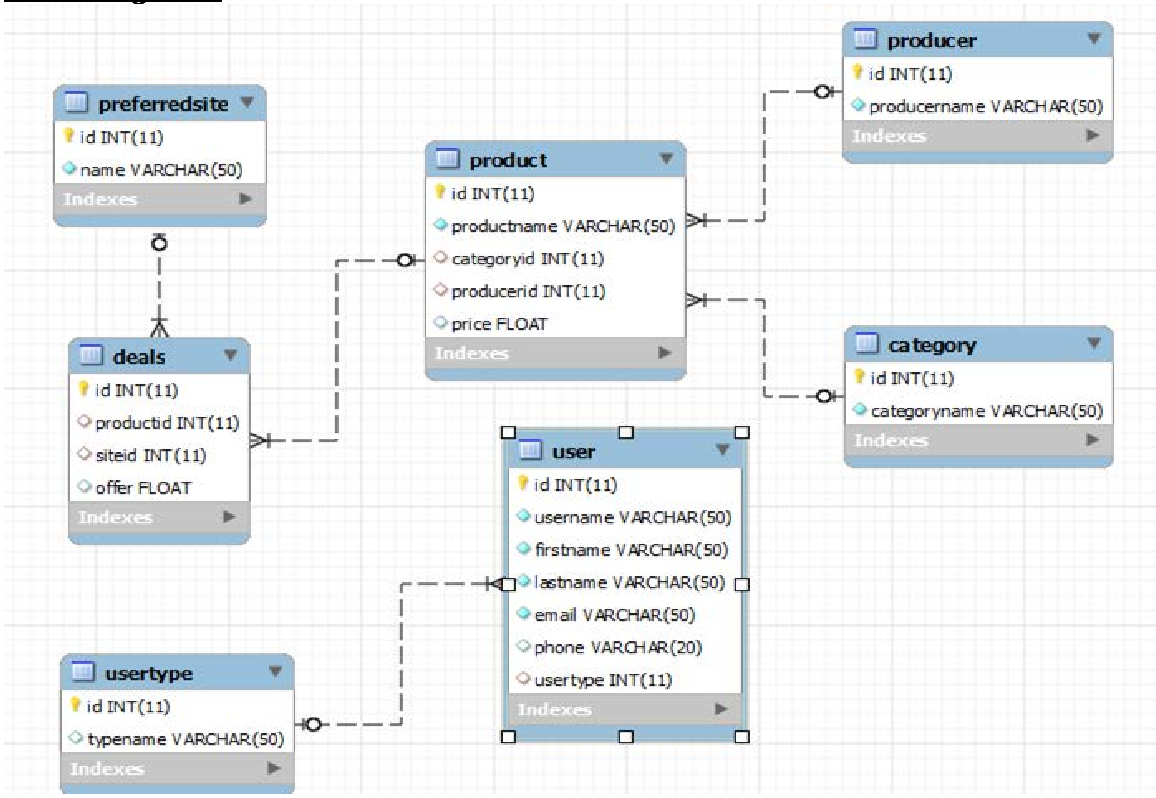
- Description: The Admin Creates Ad's to attract user attention on the website.
- Precondition:
- Step by Step:
    - Admin selects the appropriate Ad's Based on the popularity of product.
    - Admin posts the Ad's on the Website.

DESIGN:

Schema of the Database:

category(***id***, categoryname);
producer(***id***, producername);
product(***id***, productname, categoryid, producerid, price);
usertype(***id***, typename);
user(***id***, username, firstname, lastname, email, phone, usertype);
preferredsite(***id***, name);
deals(***id***, productid, siteid, offer);

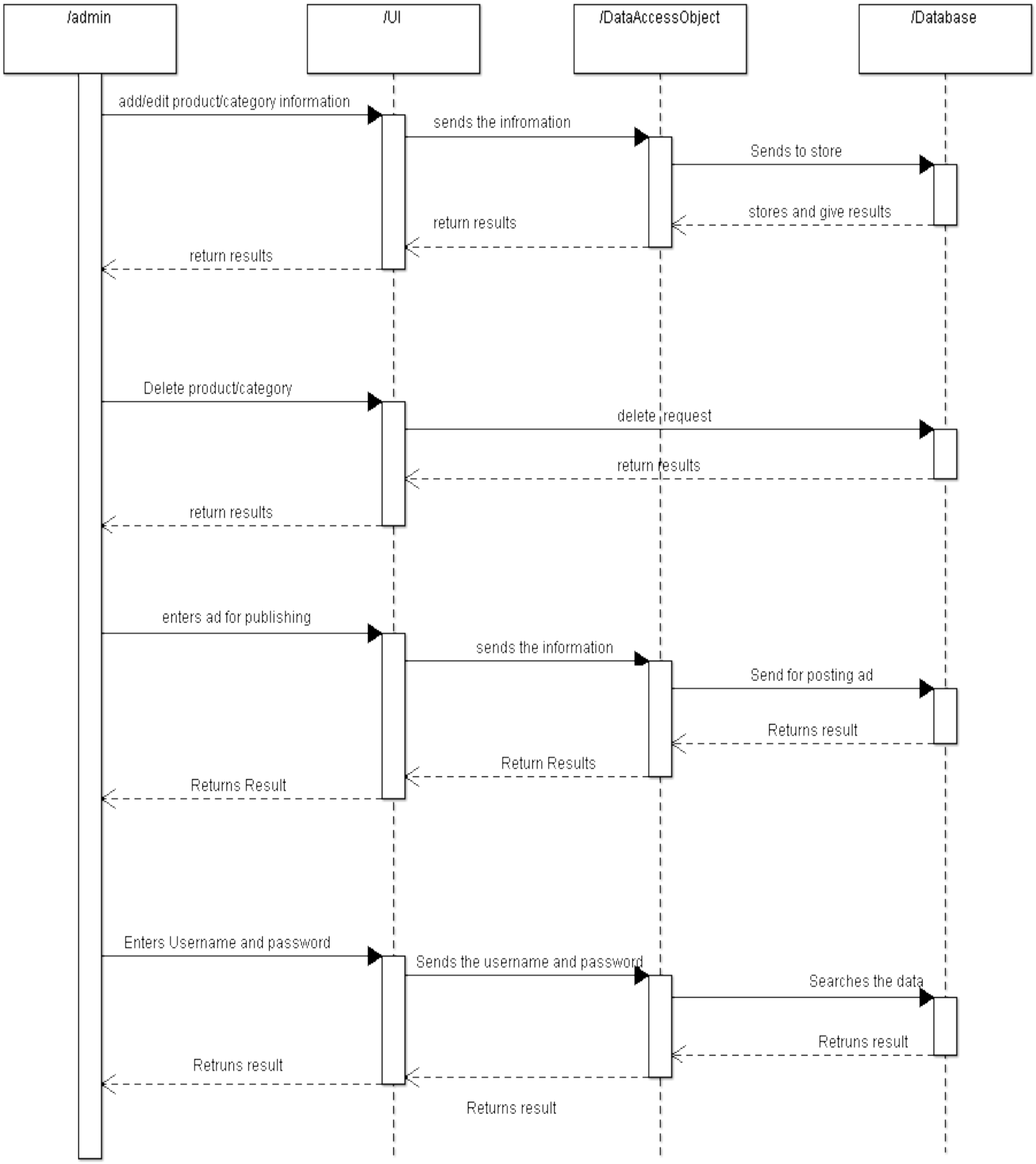
Class Diagram:



Relationship cardinality:
One user can be only one type of user, while many users can be of same type so the relation between user and user type holds a many to one relation.
Same goes with the product to category, product to producer and product to deals relations. These all are many to one relations.

The relation between deals and preferredsite is also a many to one relation because a certain site can have multiple offers for the same product.

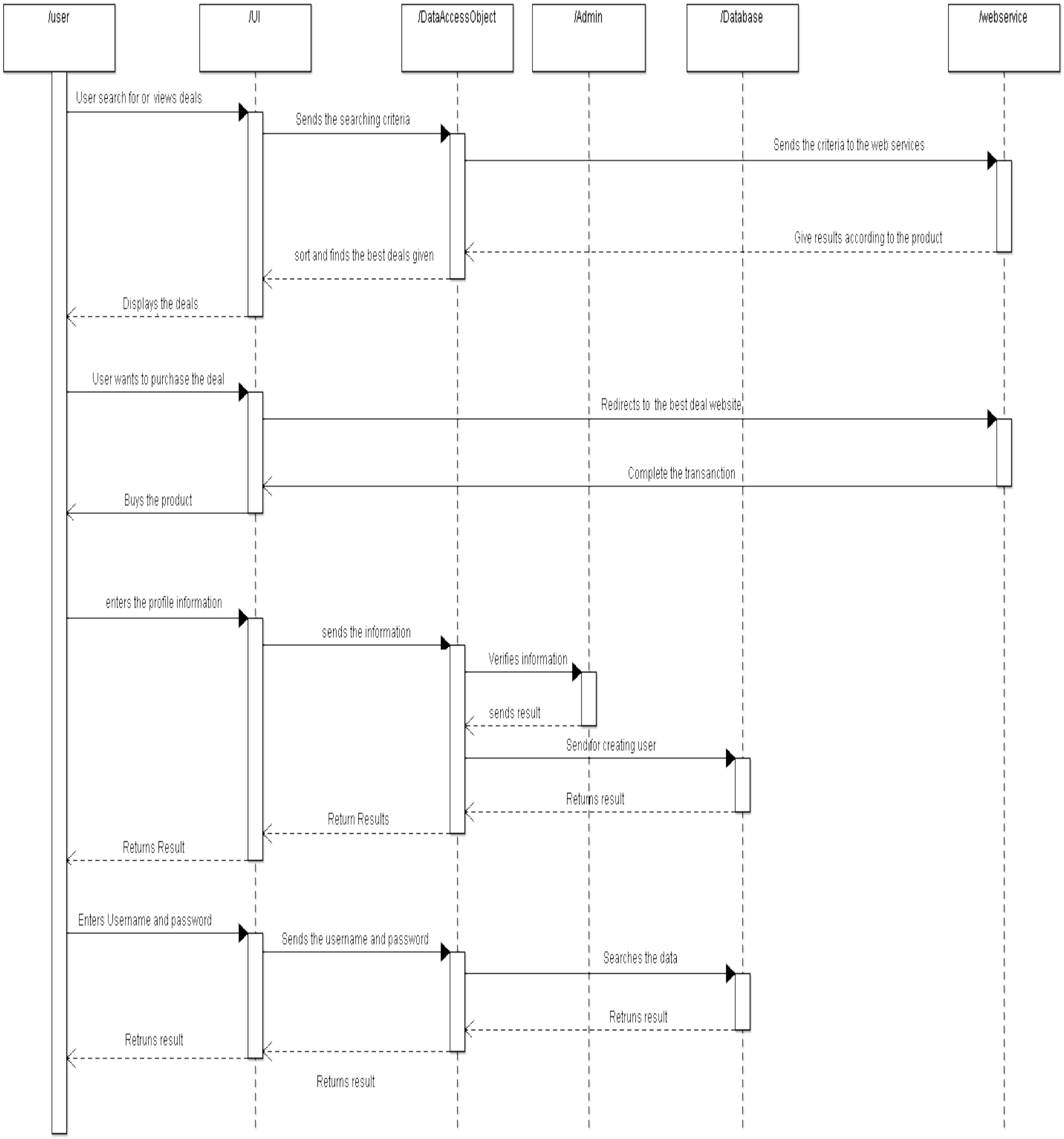
Admin Sequence Diagram:



In the above diagram the admins interaction with the application is shown. An admin can login to the site, publish ads and verify users. The step by step procedures for these actions are discussed in the use case and also shown in above sequence diagram.

User Sequence Diagram:

In the diagram the user’s interaction with the application is shown. A user can visit the site, login to the site, browse products, and purchase products. The step by step procedures for these actions are discussed in the use case and also shown in above sequence diagram.


/user
/UI
/DataAccessObject
/Admin
/Database
/webservice
User search for or views deals
Sends the searching criteria
Sends the criteria to the web services
Give results according to the product
sort and finds the best deals given
Displays the deals
User wants to purchase the deal
Redirects to the best deal website
Complete the transanction
Buys the product
enters the profile information
sends the information
Verifies information
sends result
Send for creating user
Returns result
Return Results
Returns Result
Enters Username and password
Sends the username and password
Searches the data
Retruns result
Retruns result
Returns result

Implementation:

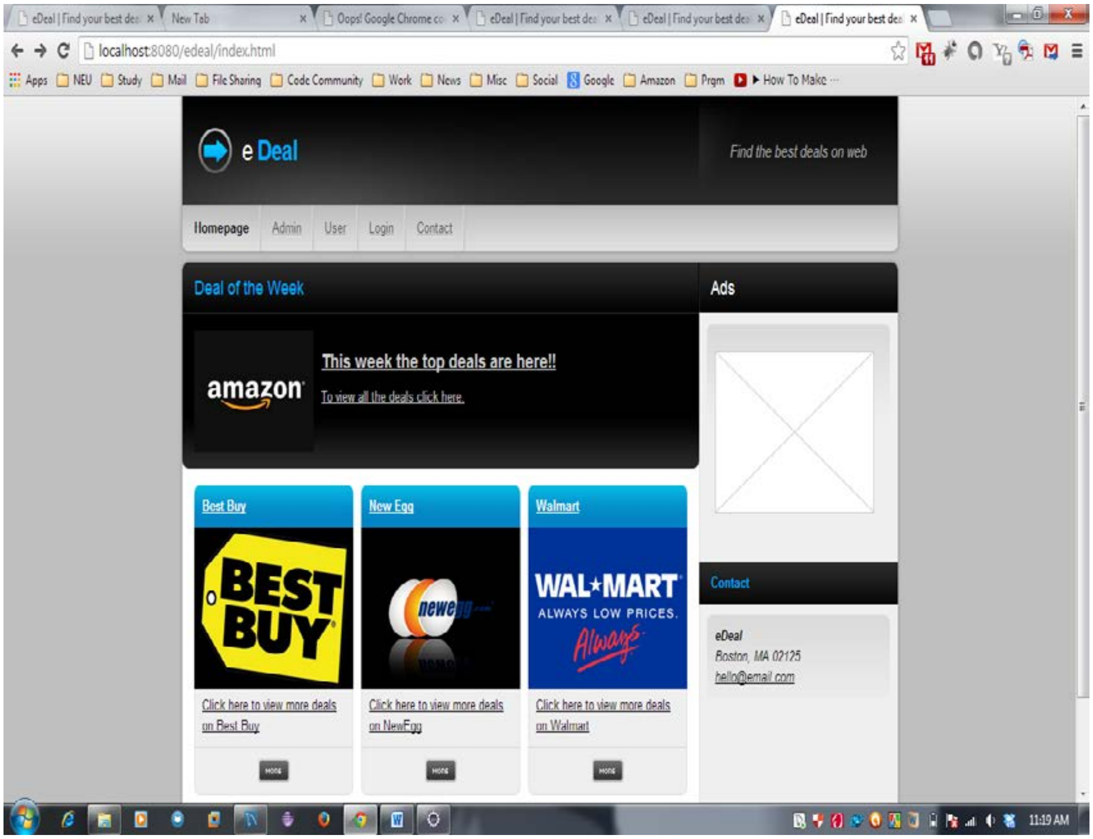
To implement our web service project we have use the following components:

- MySql Server
- Apache Tomcat server
- Eclipse IDE
- JPA
- Jackson & Jersy
- JSon

Discussion:

A web service for finding deals

Our first step to complete the project was to build a web service that gives us the best deals from different sites like Amazon, Best Buy etc.



e Deal

Find the best deals on web

Deal of the Week

This week the top deals are here!!

To view all the deals click here.

Click here to view more deals on Best Buy

Click here to view more deals on NewEgg

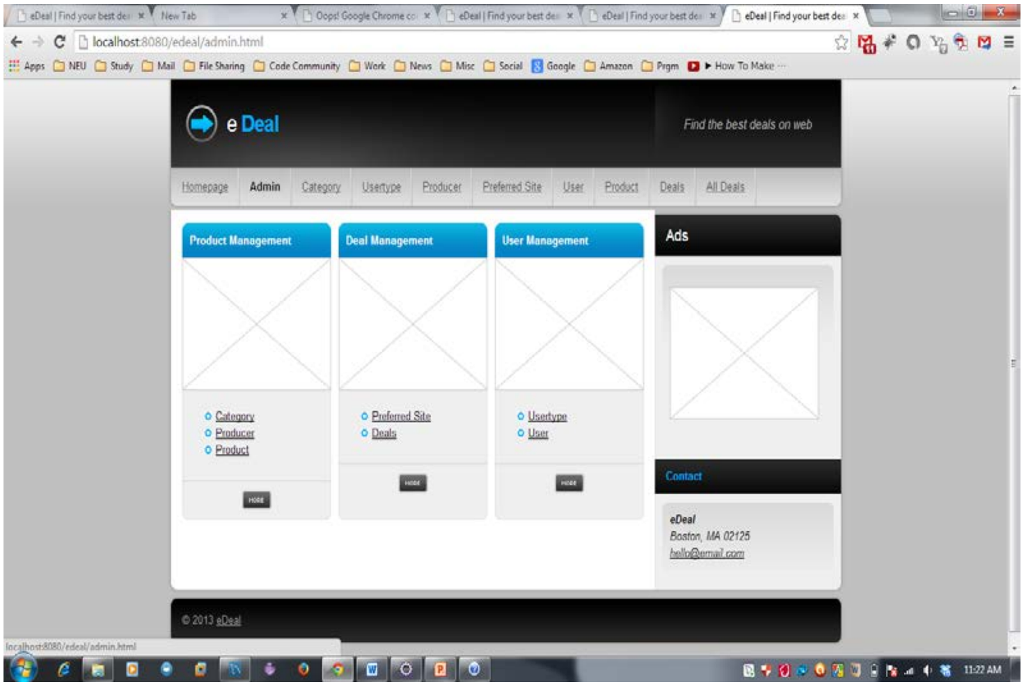
Click here to view more deals on Walmart

Contact

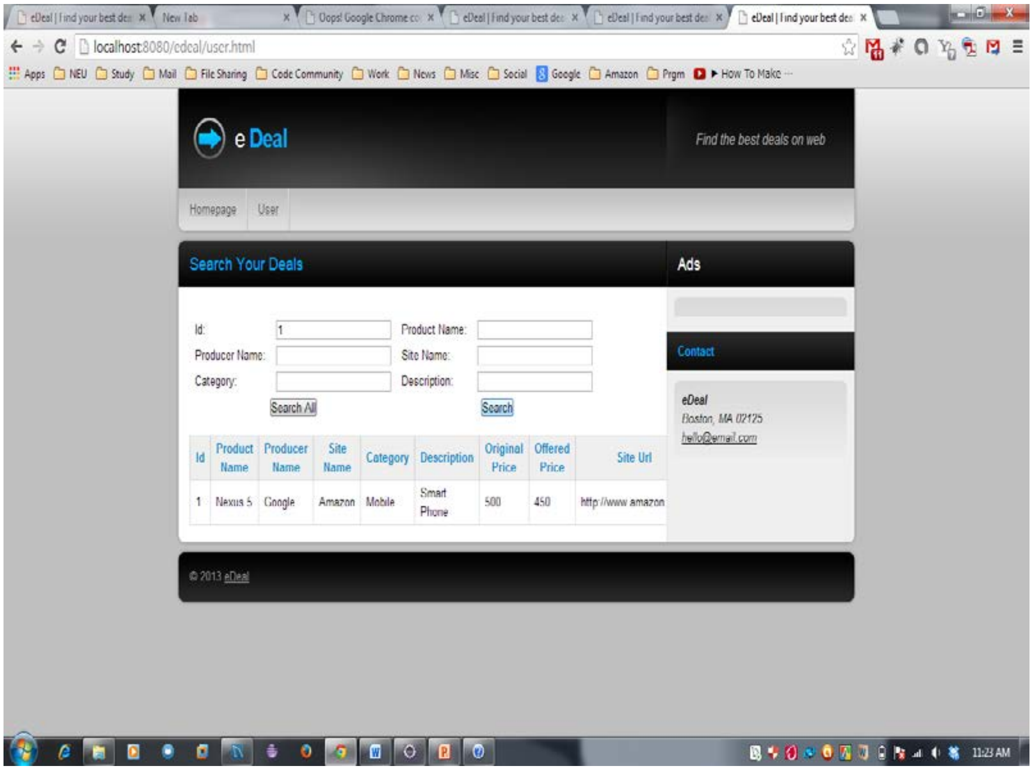
eDeal
Boston, MA 02125
hello@email.com

Login preferences:

In our application there are separate privileges for admin and user. If an admin logged in to the site he will get the following interface



If a user logs in he will get the following interface:



Search Your Deals

Id: 1
Product Name:
Producer Name:
Site Name:
Category:
Description:
Search All
Search

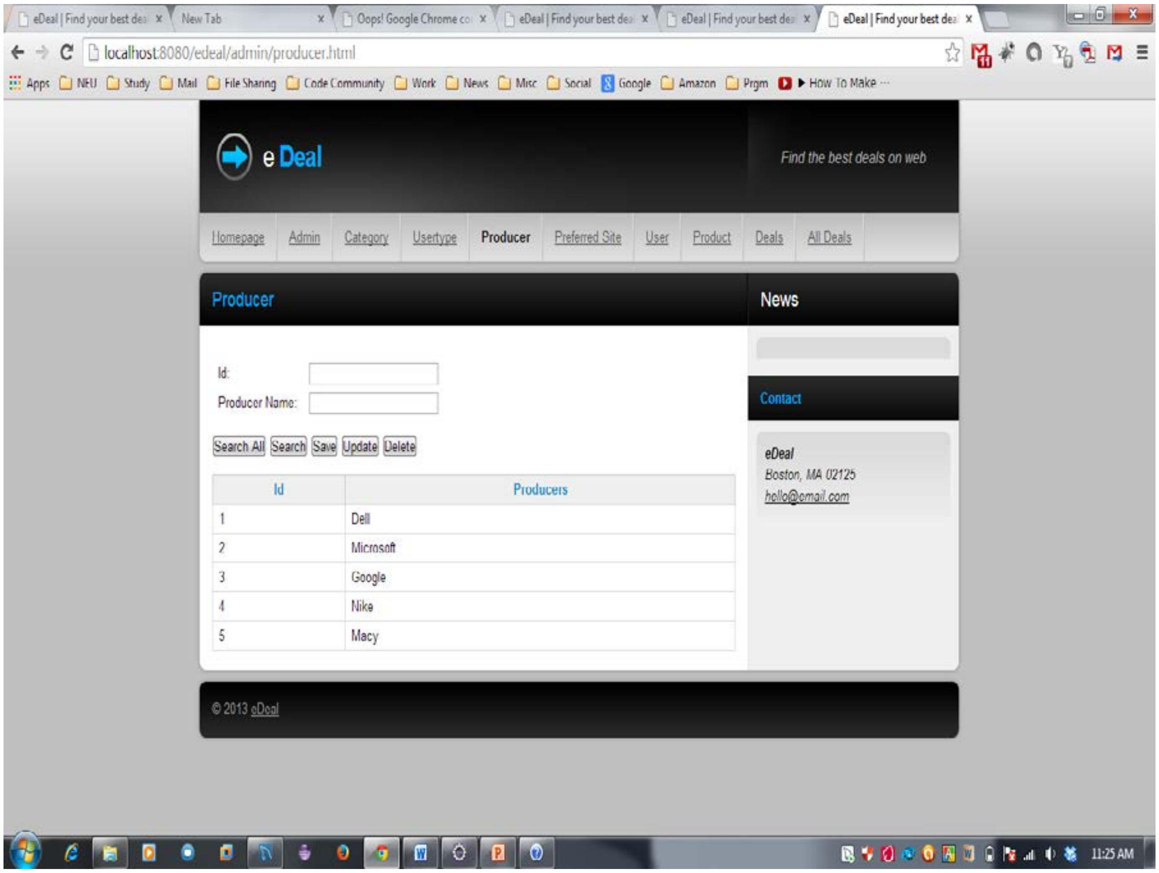
Id	Product Name	Producer Name	Site Name	Category	Description	Original Price	Offered Price	Site Url
1	Nexus 5	Google	Amazon	Mobile	Smart Phone	500	450	http://www.amazon

Search the Best Deals

Our main goal was to search the best deals available from different sites and publish them in one single page. From the above picture shown for user interface a user can search by any criteria of a product and the application will get the data from all sites and show that to the user.

Admin Actions:

Admin can make entry to the database. He can add sites to the preferred sites, validate a user, add/ update/ delete any user, add, update, delete any Producer. For example he can view the list of producers also, which will be not viewable by user like the following:



Id:

Producer Name:

Id	Producers
1	Dell
2	Microsoft
3	Google
4	Nike
5	Macy

eDeal
Boston, MA 02125
hello@email.com

In short our application gives admin the required privileges described in our requirement and restrict users to their limits and enables user to use the site as a place to find the best deals all over the web.

Conclusion:

In our project we tried to do a web service that will provide ease to the users for searching the best deals online. For the time constraint we were not able to fully design the project as we planned to do. In future we can take our project to such a level that we will be able to publish it as a perfect web service that will help the users to get their best deals at ease. Also presently available searching APIs can be included to our project to make the search more functional. We did not added any security measures in our web page, we have to implement that in future to protect our users information form threat and vulnerability.

References:

1. Sciore E. (2008). *Database Management : A Systems Approach Using Java.* Massachusets: John Wiley

2. *EclipseLink Solutions Guide for EclipseLink.* (n.d.). Retrieved from Online lecture on Eclipse Web site: http://www.eclipse.org/eclipselink/documentation/2.4/solutions/restful_jpa002.htm

3. *Getting Started with the Force.com REST API.* (n.d.). Retrieved from from developerforce website: http://wiki.developerforce.com/page/Getting_Started_with_the_Force.com_REST_API