Gear4Camp - Outdoor Gear Rental - Iteration 1



September 25th, 2016

**Table of Contents**

1. Introduction
2. Supplementary Specification
3. Glossary
4. Artifacts
5. Requirements
6. Domain Model
7. Consolidated Usecase Diagram
8. Consolidated Sequence Diagram
9. Class Diagram
10. Use Case Models
11. Database Schema
12. Testing
13. Project Management

**1. Introduction**

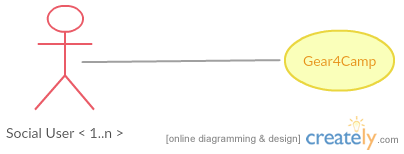
Gear4Camp is an a website which acts as online rental system for nearby users to rent outdoor equipment to each other conveniently.

In our website we will be showing following features

* Users can register for an account and become a member.
* Members can create and post Ads.
* Members can edit or delete existing Ads.
* Members can rent an item.
* Members can rate their rental transactions

**2. Supplementary Specification**

**System Perspective**



**Assumptions and Dependencies**

* Users must be able to view all available ads.
* Clicking an ad will bring the user to a new page to display more details about the ad.
* Online Payment and shipping an item are outside the scope of our system.
* Renters and buyers will contact each other and decide on payment and pick up or drop location.

**Stakeholder Perspective**

We have only one stakeholder which is common user who will be posting rent item Ads or renting items from other users.

**Tools Used**

* GitHub: Source code and documentation repository
* Eclipse: Development environment
* Trello: Project tracking and management
* Google Docs: Document Collaboration
* UML Design: Creately, Visio

**Technologies Used**

* Java, JSP, Servlets, HTML, CSS
* Server: Apache Tomcat
* Database: MySql

**3. Glossary**

|  |  |  |  |
| --- | --- | --- | --- |
| **Revision History** | | | |
| **Version** | **Date** | **Description** | **Author** |
| V 1.0 | 09/25/2016 | First draft to be refined in later versions | Group 3 |
|  |  |  |  |

|  |  |
| --- | --- |
| **Term** | **Description** |
| User | Someone who uses the Gear4Camp website without creating an account. I.e. they can view and choose ads that they may be interested in, but cannot create ads themselves or rent an item |
| Member | A user who creates an account and can create ad postings to rent out equipment. |
| Ad | Created by a member to advertise equipment available to rent. It includes various details about the equipment and various policies for renting (dates, price, etc.). |
| Home Page | Home page of the website that displays all of the created ads by users |

**4. Artifacts**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Discipline** | **Artifact** | **Inception (Iteration 1)** | **Elaboration (Iteration 1)** | **Construction (Iteration )** | **Transition (Iteration)** |
| Business Modelling | Domain Model |  | S, R |  |  |
| Requirements | Use Case Model |  | S |  |  |
|  | Use Case Text | S | R |  |  |
|  | Use Case Diagram | S | R |  |  |
|  | System Sequence Diagram |  | S |  |  |
|  | Vision | S | R |  |  |
|  | Supplementary Specification |  | S |  |  |
|  | Glossary |  | S |  |  |
| Design | Class Diagram |  | S |  |  |
|  | Interaction Diagram |  | S |  |  |
|  | Sequence Diagram |  | S |  |  |
| Implementation | Code |  | S |  |  |
| Testing | Test Case Plan |  | S |  |  |
|  | Unit Testing |  | S |  |  |
|  | System Testing |  | S |  |  |

**5. Requirements**

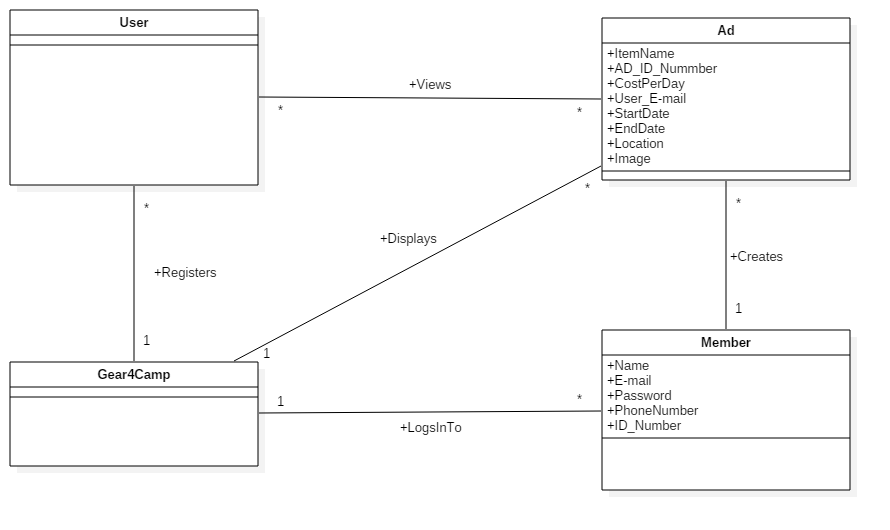
**R1 -** User can sign up to become a member. The member account will have typical information such as name and e-mail, contact number, etc. An automatically generated User ID number will be given to the account upon creation.

**R2** - Member can log into account to create a new ad for now.

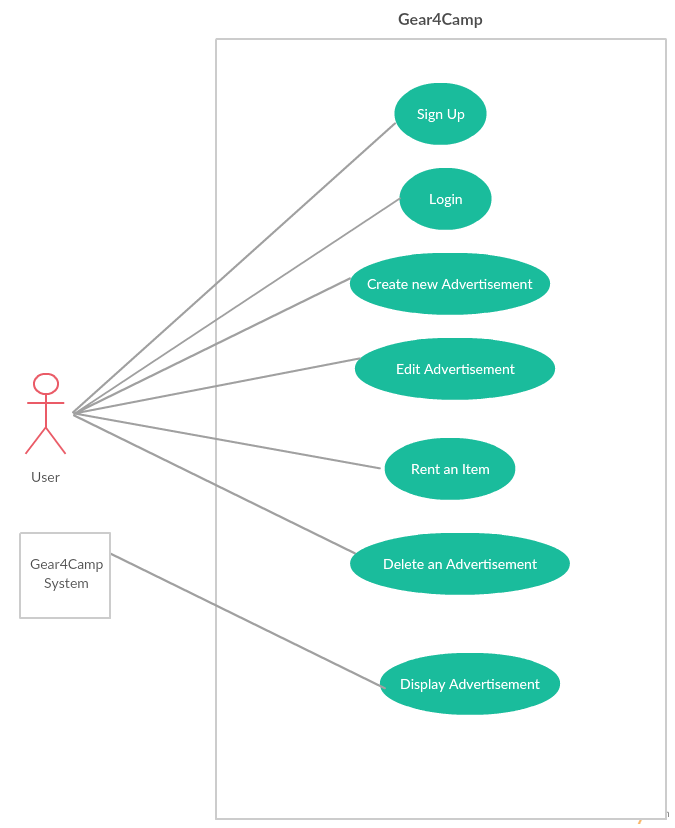
**R3** - Member can create an ad with equipment information, 1 required picture maximum only for now, required email address for contact, optional phone number for additional contact, date ranges for renting, the renting location, and requested price per day to rent. Ads will have an automatically generated and unique Ad ID number.

**R4**- User can browse equipment for rental ads on the home page.

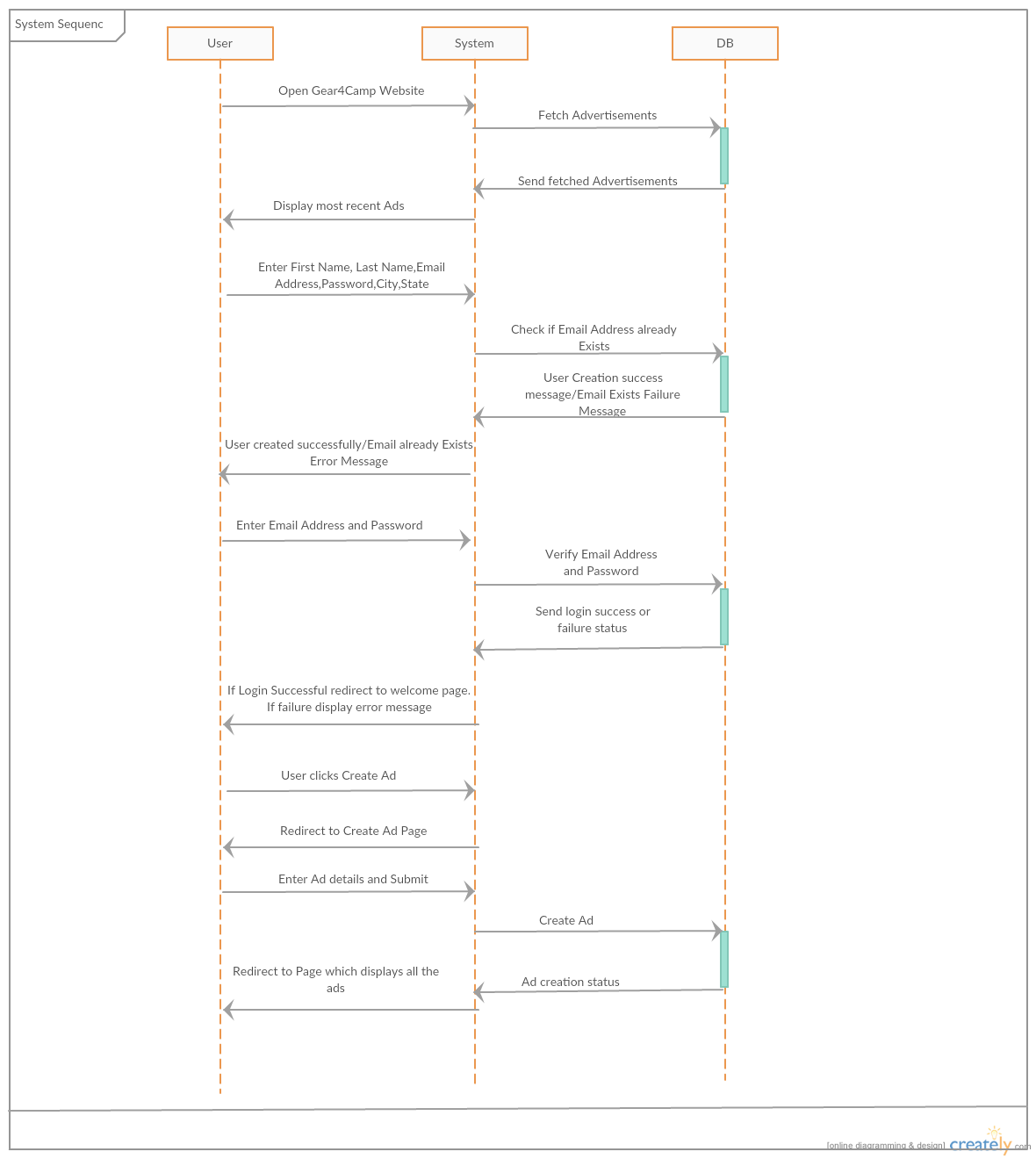
**6. Domain Model**



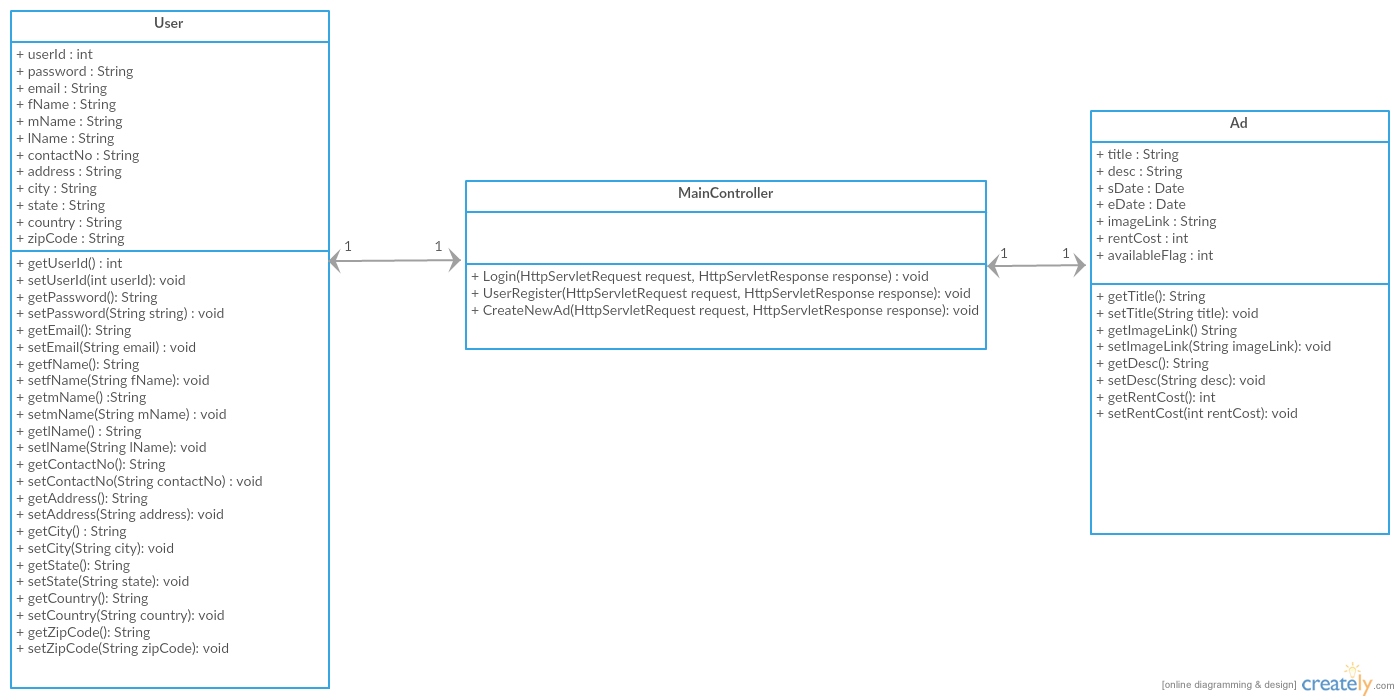
**7. Consolidated Use Case Diagram**



**8. Consolidated Sequence Diagram**

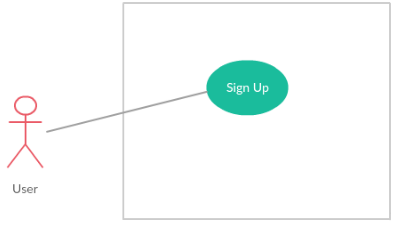


**9. Class Diagram**



**10. Use Case Model**

**Use Case UC1: Sign Up**



**Scope:** Gear4Camp application

**Level:** User goal

**Primary Actor:** User

**Stakeholders and Interests:**

**User:** wants to register in the Gear4Camp Application.

**Preconditions, Success Guarantee:** Must have valid Email address

**Main Success Scenario:**

1. User opens the website.
2. User selects to register for an account.
3. System prompts user for registration information, such as name, Email ID, password, contact number.
4. User enters his/her information.
5. System verifies information and creates account for new member.

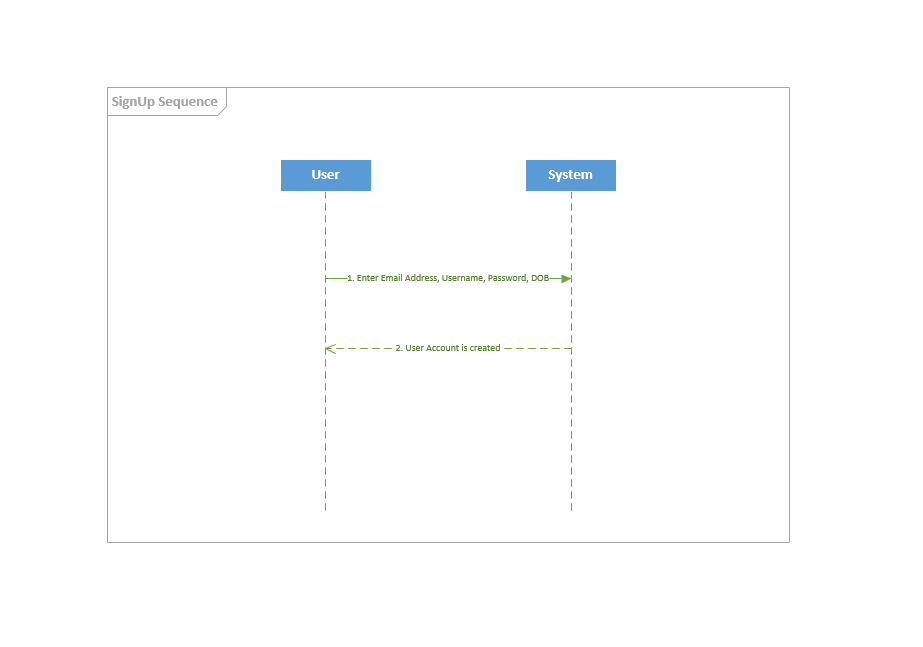
**Postcondition:** User has created an account

**Alternate Flows:**

**Alt Flow 1. Invalid Email ID**

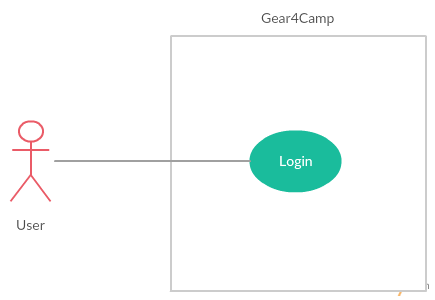
1. User enters invalid Email address (i.e. Email already used or isn’t a real Email address).
2. System detects invalid Email address and prompts user to enter a valid Email address.
3. User enters valid Email address and continues registration.

**Use Case Sequence Diagram:**



**Use Case UC2: Login**

**Use case Model:**



**Scope:** Gear4Camp application

**Level:** User goal

**Primary Actor:** User (Primary)

**Stakeholders and Interests:**

**User**: Wants to be able to log into account

**Preconditions, Success Guarantee:** User has registered for an account (UC1).

**Main Success Scenario:**

1. User opens the Website and is not logged in the system.
2. System requests that the user enter his/her username(Email Id) and password.
3. User enters his/her username and password.
4. System validates the entered username and password and logs the user into the system.

**Postcondition:** User is logged into account

**Alternate Flows:**

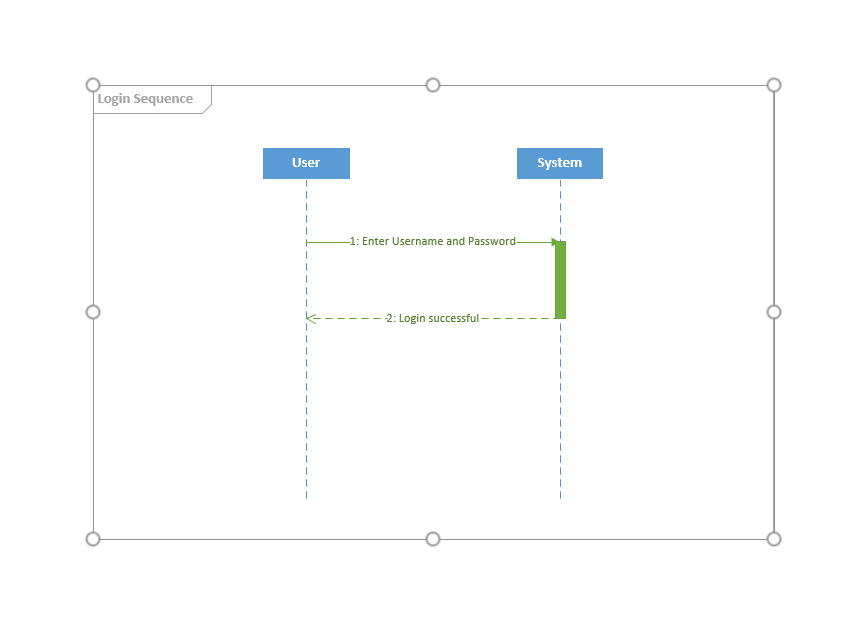
**Alt1. Invalid Username / Password**

1. User enters an invalid username and/or password
2. System displays an error message.

**Special Requirements:**

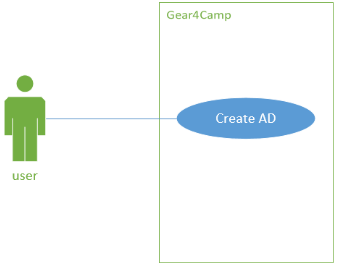
Unique Username

**Use Case Sequence Diagram:**



**Use case UC3: Create AD**

**Use case Model:**



**Scope:** Gear4Camp application

**Level:** User goal

**Primary Actor:** User

**Stakeholders and Interests:**

**User**: wants to post an advertisement on Gear4Camp website.

**Preconditions, Success Guarantee:** User has a valid account and successfully login into the website.

**Main Success Scenario:**

1. User click the create AD button in the web page.
2. System direct user into AD creation page and provide text field and picture upload button for user.
3. User edit AD content in provided area and upload pictures if necessary.
4. System save the AD content into database and post it in the website.

**Postcondition:** User has created an ad.

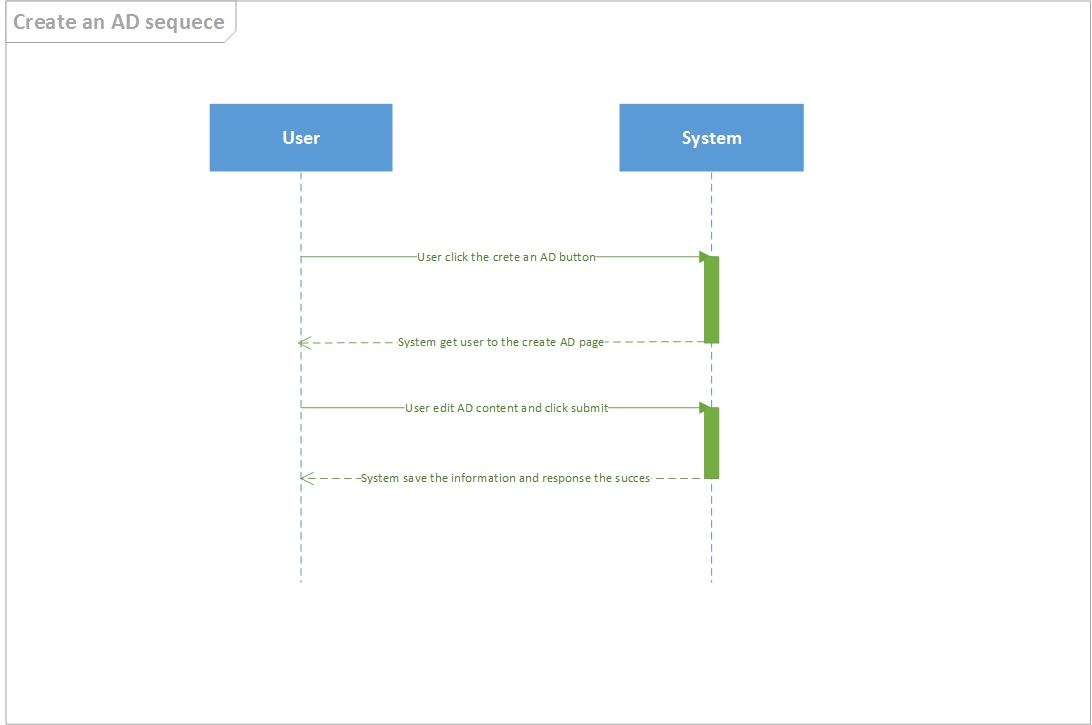
**Alternate Flows:**

**Alt1. Create AD without login**

1. User click the create AD button in the web page.
2. System direct user into AD creation page and provide text field and picture upload button for user.
3. User edit AD content in provided area and upload pictures if necessary, as the user is not login, he/she should enter contact information in the provided area.
4. System save the AD content into database and post it in the website.

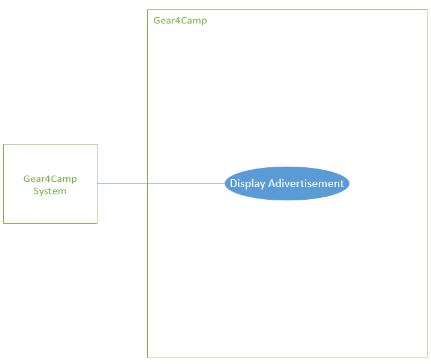
**Special Requirements:** N/A

**Use Case Sequence Diagram:**



**Use case UC4: Display ADs**

**Use case Model:**



**Scope:** Gear4Camp application

**Level:** User goal

**Primary Actor:** User

**Stakeholders and Interests:**

**User**: wants to browse some spare item posted in the website which he/ she is interested.

**Preconditions, Success Guarantee:** User opens the Gear4Camp website

**Main Success Scenario:**

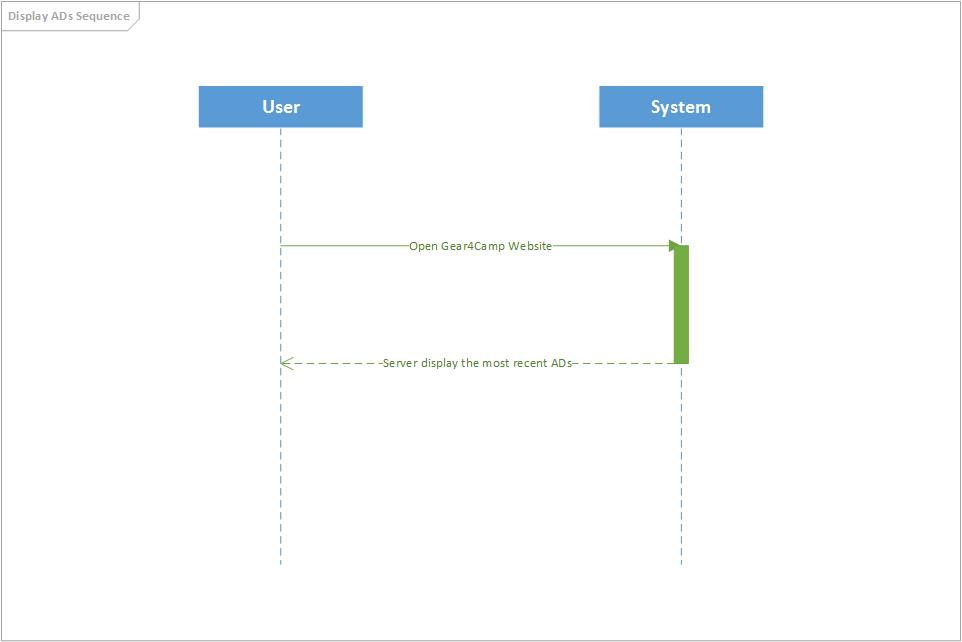
1. User is the main page of Gear4Camp
2. Server grab the latest ADs and display the key information, including title, item type provide name, location, etc.
3. User can view all created ads.

**Postcondition:** Users are able to view created ads

**Alternate Flows:** N/A

**Special Requirements:** N/A

**Use Case Sequence Diagram:**

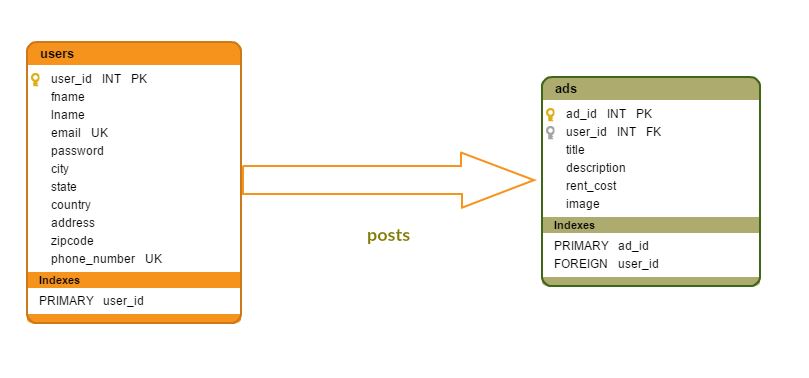


**11. Database Schema**

For Iteration 1 we have 2 tables in our database.

1. Users: This database has the details of all the users like First name, Last name, Email ID, phone number, address and more
2. Ads: This database has details of all the ads which are present right now. For each ad the details such as Title, description, image, cost and many more.

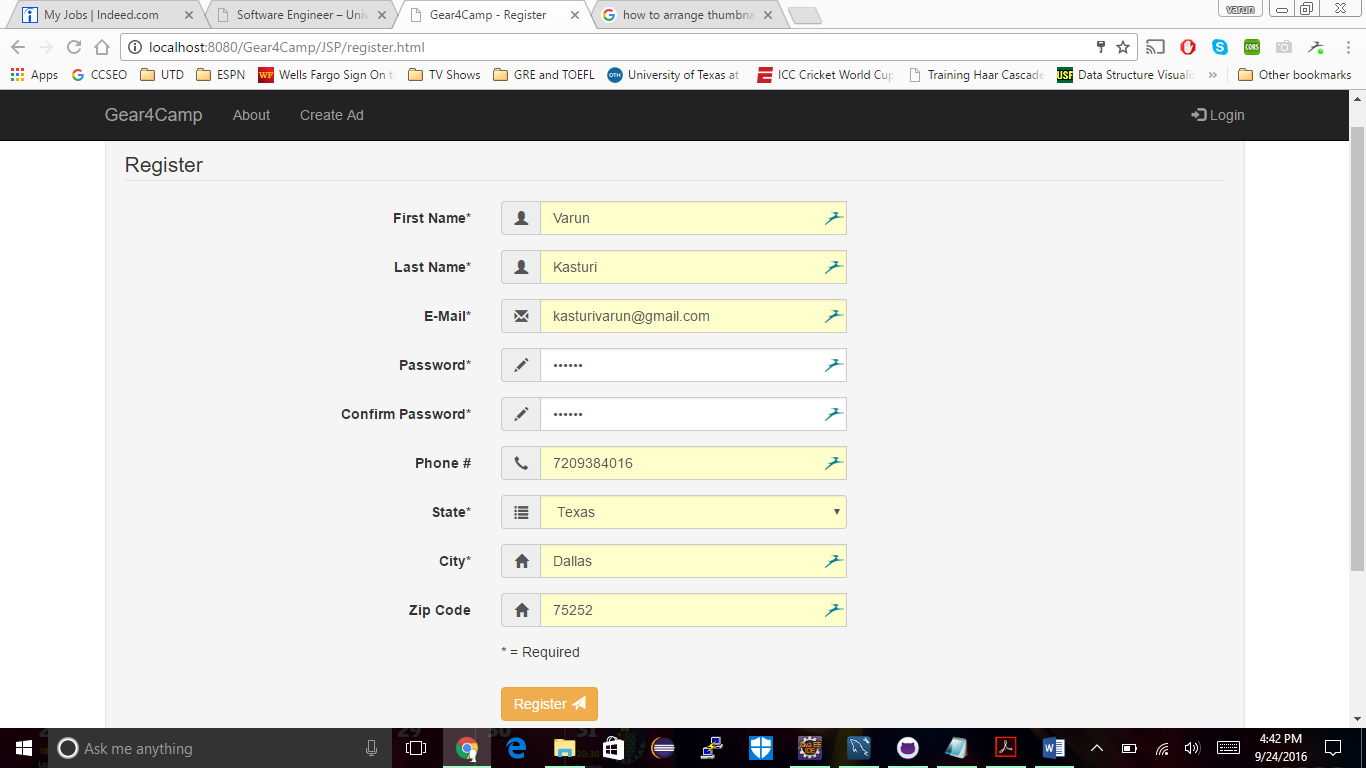
When user posts an ad it will stored in the database.



**12. Testing**

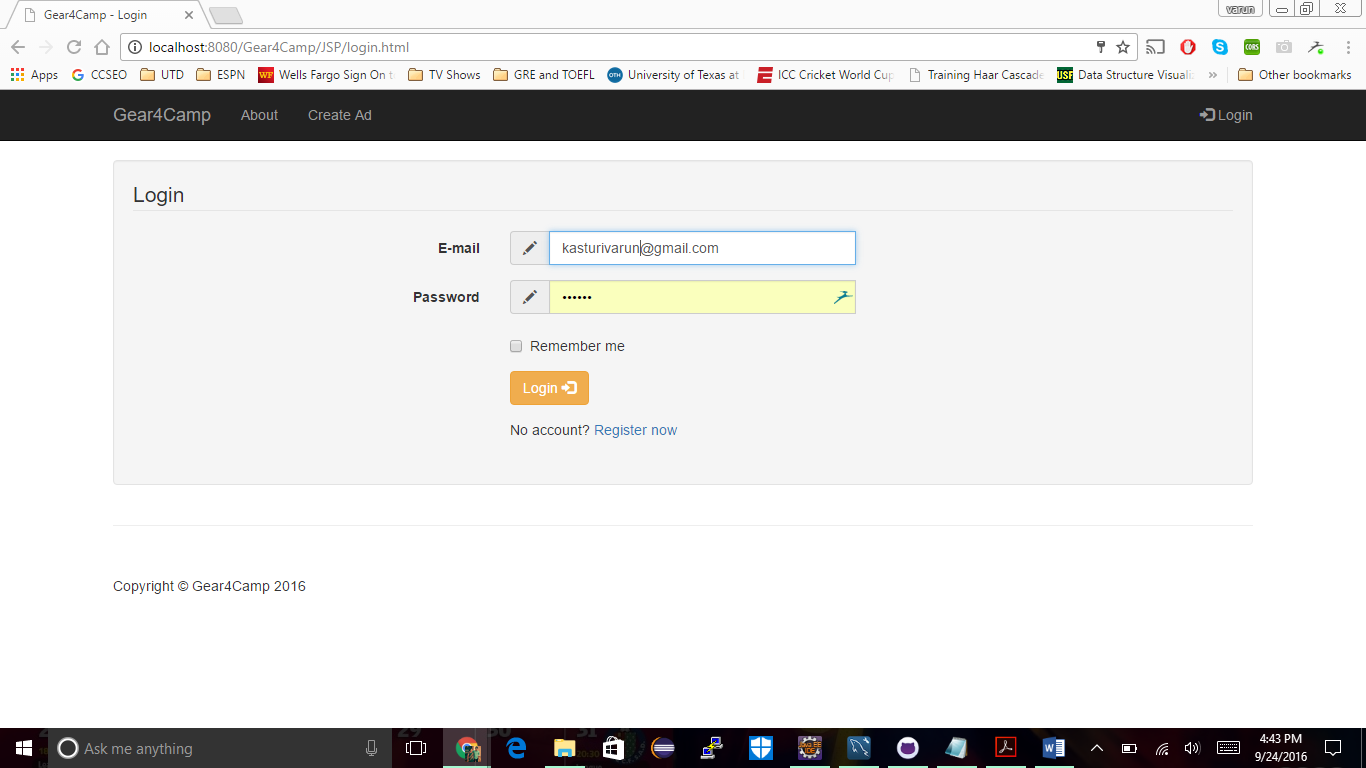
**Testing UC1: Sign Up**

We submitted test information into the register form. The information was successfully stored in the database and the account was created.



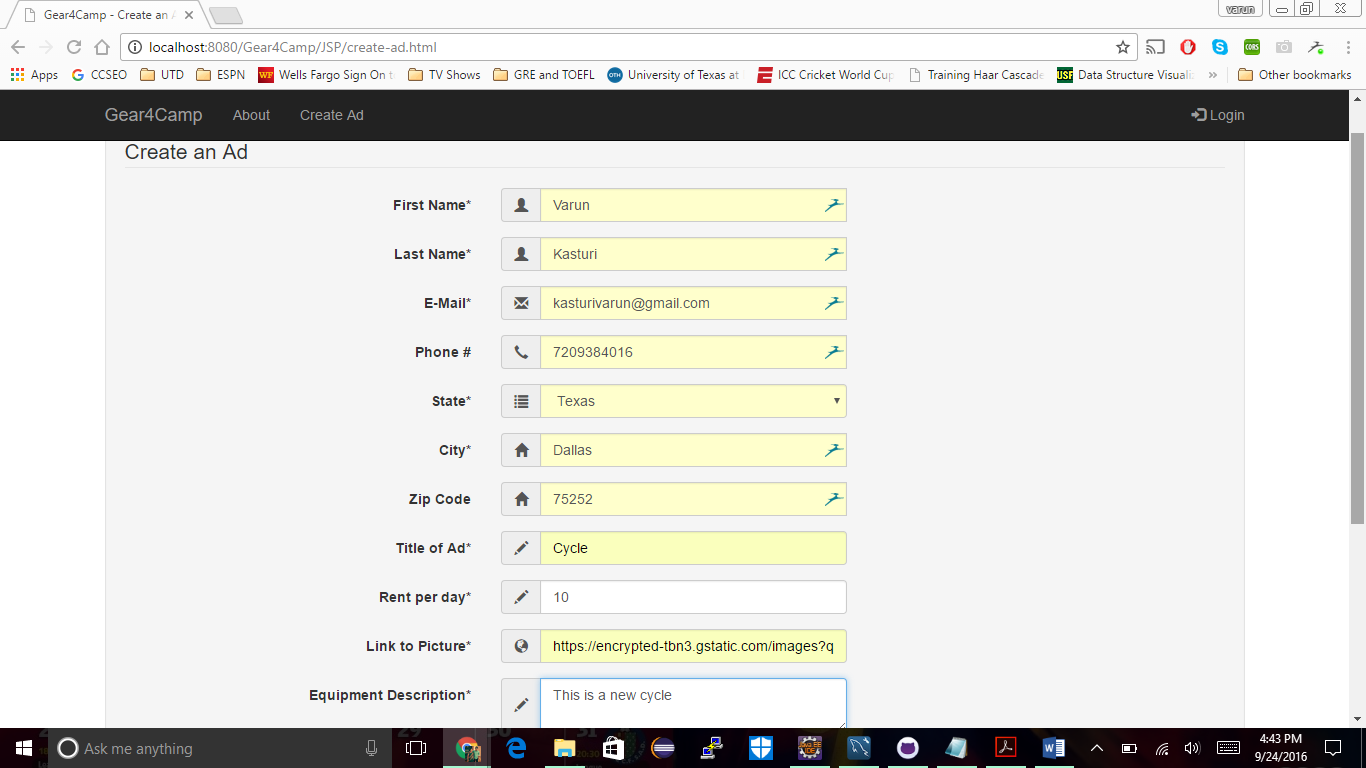
**Testing UC2: Login**

We tested logging into the dummy account made previously. The application was able to check and verify the submitted login details with the database and login was successful.



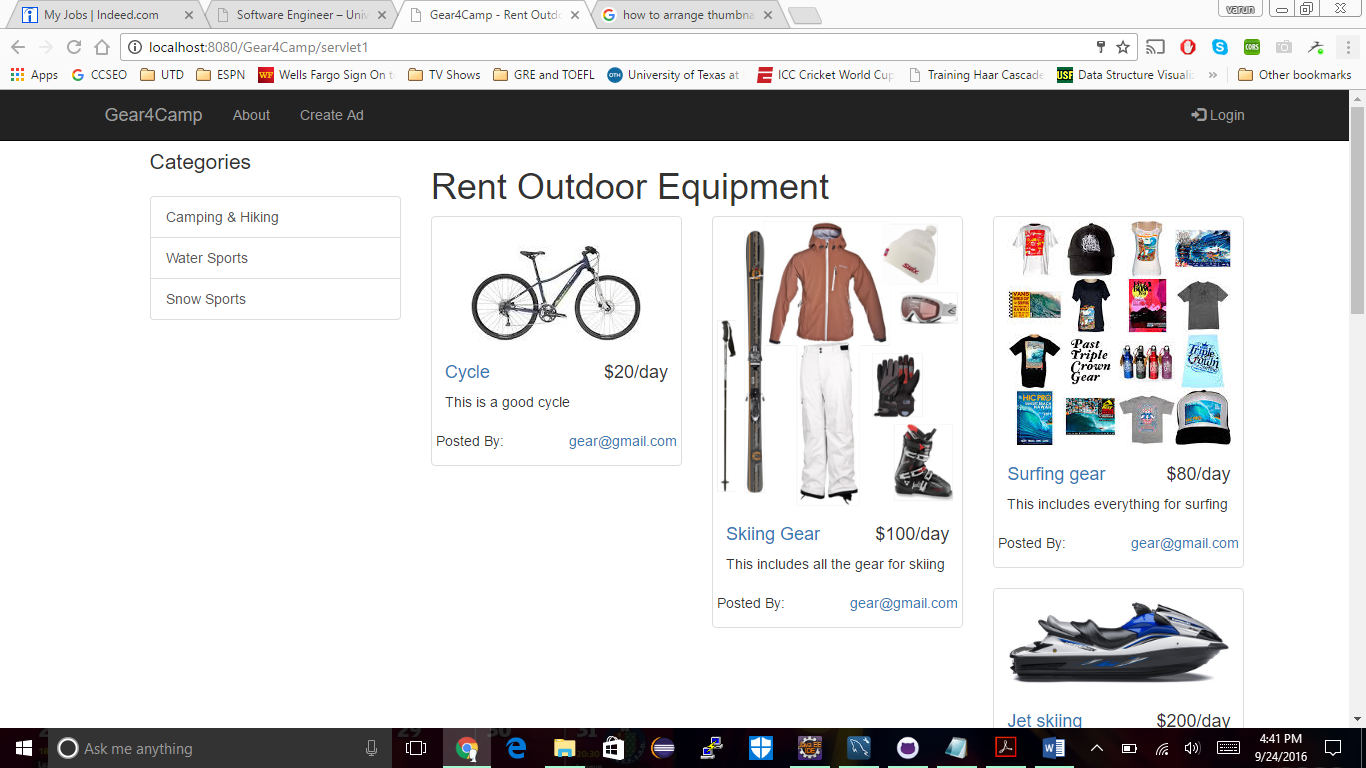
**Testing UC3: Create Ad**

After logging in, we submitted information to create an ad. The data was stored successfully on the database and the ad was created successfully.



**Testing UC4: Display Ads**

After creating an ad, the application pulled the ad information from the database and displayed on the homepage successfully.



**13. Project Management**

**Expectations for Iteration 2:**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Date** | **ID** | **Task** | **Assigned** | **Dependency** | **Comments** |
| 09/25/16 | 2.1 | Use cases for user ad history and logging out | Varun, Sneha |  |  |
| 09/25/16 | 2.2 | UML Diagrams for user ad history and logging out | Varun, Sneha |  |  |
| 10/15/16 | 2.3 | Implementation for user ad history and logging out | Varun, Sneha | 1.1,1.2 |  |
| 10/15/16 | 2.4 | Use case for searching, filtering, and renting an item | Henry, Kainan |  |  |
| 10/15/16 | 2.5 | UML Diagrams for searching, filtering and renting an item | Henry, Kainan |  |  |
| 10/20/16 | 2.6 | Implementation for searching, filtering and renting an item | All | 1,4,1.5 |  |
| 10/18/16 | 2.7 | Milestone: Iteration 2 | All |  | Members may view their accounts to display all of their ad postings and logout. The application will allow searching and filtering for ads. |