# Iteration 1 Feature Set

Snap-2-Ask

## Web (Tutor)

**Registration (Form for submitting information necessary to create an account)**

GUI:

* Display a friendly message if anything goes wrong or if any fields are invalid
* Personal Information
  + First Name
    - Not empty
  + Last Name
    - Not Empty
  + Email Address
    - Not Empty
    - Valid Email Address
* Payment Information
  + Credit Card Number
    - Valid Credit Card Number

Backend:

* Create the REST API function for creating a new user account
* Create the session token upon successful creation of the user account
* Redirect the user to the main browsing page of Snap2Ask after the account is created

**Feature : Login (Form for submitting email and password in order to sign in to Snap2Ask)**

GUI

* Facebook
  + Button to connect with your Facebook account
* Google
  + Button to connect with your Google+ Account
* Snap Account
  + Email Field
    - Must not be empty
    - Must be a valid email
  + Password Field
    - Must not be empty

Backend:

* Verify the user credentials if using Snap2Ask’s custom authentication
* Call the respective methods in the Google/Facebook API’s if the user selected the Facebook/Google login.
* Display any errors on the same page as the login screen
* If the authentication was successfully, set the session token and redirect to the main page

**Logout**

GUI:

* Will sign the user out of Snap2Ask and return the the main login/registration page

Backend:

* Clear the session variables

**Tutor Validation**

GUI:

* A form for a tutor to specify a preferred subject
* Will have a simple test of the user’s knowledge in order to validate the tutor

Backend:

* Verify the tutor validation test using the database
* Add the preferred subject to the user’s account if the validation was successful.

**View/Edit Account**

GUI:

* View/Edit Personal Information
  + View Email Address
  + View First/Last Name
  + Modify First/Last Name
* View/Edit Payment Information
  + View credit card information
  + Add/Edit/Remove credit card information

Backend:

* Get the information from the REST API using a GET call
* Update the information using the REST API with a PUT call

**Delete Account**

GUI:

* A button to delete the user’s account from Snap2Ask

Backend:

* Implement a REST API function to delete the specific account from the database.

**View Questions**

GUI:

* View Recent Questions
  + Questions will be displayed for the tutor categorized by date.
  + Most recent questions will be displayed first
* View By Academic Category
  + Questions will be displayed for the tutor categorized by academic subject
  + If the user selects “Math”, only “Math”-related questions will be displayed

Backend:

* Implement a REST API function to retrieve a list of all the questions sorted by date
* Implement a REST API function to retrieve a list of questions filtered by a specific subject

**Answering Questions**

GUI:

* Details of the current question will be displayed
  + Image
  + Category
  + Other meta-data (E.G. date)
* Tutor will be able to submit an answer to the question
  + Textual answer
  + Must not be empty

## Backend:

* Implement a REST API function to post the answer to the question
* Send the student who asked the question a push notification informing them that the question has been answered

## iOS (Student)

**Registration (Screen for submitting information necessary to create an account)**

GUI:

* Display a friendly message if anything goes wrong or if any fields are invalid
* Personal Information
  + First Name
    - Not empty
  + Last Name
    - Not Empty
  + Email Address
    - Not Empty
    - Valid Email Address

Backend:

* Create the REST API function for creating a new user account
* Create the session token upon successful creation of the user account
* Redirect the user to the main browsing page of Snap2Ask after the account is created

**Login (Form for submitting email and password in order to sign in to Snap2Ask)**

GUI:

* Facebook
  + Button to connect with your Facebook account
* Google
  + Button to connect with your Google+ Account
* Snap Account
  + Email Field
    - Must not be empty
    - Must be a valid email
  + Password Field
    - Must not be empty

Backend:

* Verify the user credentials if using Snap2Ask’s custom authentication
* Call the respective methods in the Google/Facebook APIs’ if the user selected the Facebook/Google login.
* Display an errors as alerts windows
* If the authentication was successful, then move the user to the application’s main page.

**Logout**

GUI:

* A simple logout button

**Ask Question**

GUI:

* Take/Select Photo
  + Optionally retake the photo or choose a different photo
* Select Subject (E.G. Math/Science)
* Input optional description

Backend:

* Implement a REST API function to upload the image metadata into the database
* Implement a REST API function to upload the actual image into the Snap2Ask Amazon S3 bucket named according to its question\_id.

**View Question History (A list of all previously asked questions, along with their answers)**

GUI:

* View Unanswered Questions
  + These questions have not been answered before
* View Answered Questions
  + These are all the questions that have already been answered
* View All Questions
  + This is a list of all of the questions that the user has previously asked.

Backend:

* Implement a REST API function to get a list of all questions
* Filter the list of questions according to which option the user selects to view the questions
  + Unanswered/Answered/All

**View Specific Question**

GUI:

* View the image
* View the subject
* View the description
* View Discarded Answers of the question
* View new answers to the question

Backend:

* Implement a REST API function to get the information of a specific question