Software Requirements Specification

for

BITSMarket

Version 1.0 approved

Prepared by

Shantanu Prakash (2012C6PS696P)
Vivek Kishore (2012C6PS650P)
Nirant Kasliwal (2012C6PS694P)
Arjun KV (2012A7PS030P)
Shreyansh Bajaj (2012A7PS088P)

November 25th, 2014

Table of Contents

Revision History

Name	Date	Reason For Changes	Version
Version 1	12th Nov.	Category fields, improvement on UI.	1.0

1. Introduction

1.1 Purpose

The primary goal of this Software Requirements Specification (SRS) is defining and describing the functions and specifications of **BITSMarket**, a Client to Client E-Commerce Portal. This Software Requirements Specification illustrates, in clear terms, the system's primary uses and required functionality as specified by our customer.
The online shopping software main purpose is to provide customers with the possibility to perform online purchases on products already on store.

1.2 Document Conventions

Specific terminology is used throughout the specification of the system.

User Profile: stands for the profile of the customer (person) opened in the software. One person can have multiple profiles using different emails. A profile can be linked to none or one account type.

Person: defines a real person who has an identity defined by class attributes. A person can have multiple profiles and consequently multiple accounts. For instance a person can have a basic and a business account.

Account: defines an entity that enables the user to operate throughout the system and perform purchases.

Seller: defines a person who already has a registered account to act as a seller and upload products. He also has options to view his products, make changes or delete the products.

1.3 Intended Audience and Reading Suggestions

This document is intended for the use of people involved in the development of an online product cataloging system. Here, the classmates of *Software Engineering (IS F341)* are our intended audience along with the faculty **Dr. Yashvardhan Sharma**. The document contains the specifications of the domain and functionalities of our software.

Rest of the document contains an overview of the product's description, perspective and general features. It is suggested that the reader reads the overall product description to get an idea of the various functionalities. Then it's suggested to go through the features to get an idea of the product's main features. It is also suggested to go through the software and hardware requirements.

1.4 Project Scope

The software system being produced is called Bits Market. It is being produced for a customer interested in selling via the Intranet. This system is largely cross-platform and is available to anyone using the BITS Server. The system will be run on a central server with each user having a remote user interface through a web browser to interact with it. Anyone using the BITS Server can view the available products and obtain the seller information for the same. To sell the user has to create a membership account. The member can put up the details of any items they wish to sell.

The aim of this project is to promote an efficient, user-friendly, time-fashionable, safe way for BITSians to buy and sell products (like books, electronics and others) through an easy catalogue-like platform.

1.5 References

- Twitter's CSS bootstrap
  http://getbootstrap.com/2.3.2/
- Python's web framework Django's documentation and tutorials
  https://www.djangoproject.com/
- SRS IEEE template
  www.uni-obuda.hu/users/boraros-bakucz.andras/2013/srs_template.doc
- Use case example
  www.itq.ch/tools/use_case_template.doc

2. Overall Description

This section includes details about what is and is not expected of the BitsMarket in addition to which cases are intentionally unsupported and assumptions that will be used in the creation of the BitsMARKET.

2.1 Product Perspective

BitsMarket is an online C to C website to facilitate commerce activities between BITSIAN students. The website must be available to anyone using the BITS Server and work correctly in both Internet Explorer and Mozilla Firefox. There are no hardware or software requirements beyond these including, but not limited to, memory or specific software packages that need to be utilized nor software packages that need not be utilized.
Since the product is nothing more than a web app enabling users to access as customers or sellers,

2.2 Product Features

Bitsmarket will provide a number of functions; each is listed below.

- Maintain records for many members
    1. A member has a username (unique across all users), email address (no restrictions), and BITS address (no restrictions).
    2. Any student from BITS sign up for a membership account
    3. Members can modify the profile any time they want

- Putting up Products for sale
    1. Only members of the website can put up products for sale
    2. Products have the following
        a) Product Name
        b) Category
        c) Product Description
        d) Product Image

• Show a listing of items on sale
    1. The product appear on the home page in a grid which can be viewed by any user
    2. The products are displayed in ascending order by the category
    3. The following products details are available
        a) Product description
        b) Product image

- View Seller details
    1. On the selection of any product the following details can be viewed by the users
        a) Product
            i. Product name
            ii. Product description
            iii. Product image
        b) Seller
            i. Seller Name
            ii. Local address
            iii. Phone number
            iv. Email address

- Admin
  Admin can manage the user accounts, products and categories
        a) User
            i. Can remove unwanted users
            ii. Can monitor user activity
            iii. Can create new users with various levels of permissions
        b) Products
            i. Can remove unwanted product listing
            ii. Can remove expired product listing
        c) Categories
            i. Add ,Delete or Modify categories

• Allow customers and managers to log in and out of the system.
• Users (both customers and the manager) will be logged out if inactive for 30 minutes.
• Shopping cart
• Anyone is able to add one or more books to the shopping cart.
• The shopping cart does not need to allow multiple copies of any book.

2.3 User Classes and Characteristics

Following are the three classes of users:

Buyer: The typical BitsMarket user, as a buyer, is simply anyone that has access to the Internet and a web browser in BITS. It is assumed that the user is familiar enough with a computer to operate the browser, keyboard and mouse and is capable of browsing to, from and within simple websites. Basic usage of web browsing is required for the buyer.

Seller: The typical BitsMarket user is simply anyone that has access to the Internet and a web browser in BITS. It is assumed that the user is familiar enough with a computer to operate the browser, keyboard and mouse and is capable of browsing to, from and within simple websites.Basic usage of web browsing

Administrator: The typical BitsMarket user is simply anyone that has access to the Internet and a web browser in BITS. It is assumed that the user is familiar enough with a computer to operate the browser, keyboard and mouse and is capable of browsing to, from and within simple websites.Basic usage of web browsing

2.4 Operating Environment

The software is a web app hosted on BITS LAN. The hosting can be done either on an Apache server with mod_wsgi support or using django's development server itself. It'll operate on any OS which supports a web browser. Tried on Mac OS, Windows and Linux. It is totally cross platform, its compatible on Mozilla firefox, Netscape navigator, Google chrome and internet explorer.

2.5 Design and Implementation Constraints

The security is not a concern for this system. The database may
store passwords in plain text and there is no lockout after numerous invalid login attempts. As such, the system may not work correctly in cases when security is a concern.

2.6 User Documentation

Since the product and its usage is pretty straightforward, there was no requirement for tutorials on usage and/or handbook. Anyone who has ever used or is familiar with nternet web pages, should be comfortable.

An FAQ is planned to be put up, for helping new users understand the usage and functionalities of the webapp.

Assumptions and Dependencies

Client:
We have assumed user is capable of operating these system's basic functions including but not limited to being able to power on the system, login and open either Internet Explorer or Mozilla Firefox, and navigate the browser to the address of this Bitsmarket website.

Provider:
We have assumed that the BitsMarket will be running on a properly working web server and database system with an Internet connection that allows this system to perform all communications with clients.

3. System Features

3.1 Categorization of commodities for sale.

3.1.1 The products are categorized depending on the administrators discretion. The need of different categories may evolve overtime and hence the flexibility of making and changing categories is essential.

3.1.2 Stimulus/Response Sequences

Too many products having some common feature going under the Other category is the stimulus and the response involves the creation of a category based on the common feature.

2.1.3 Functional Requirements

REQ-1:Should have access to the admin site(i.e have an administrator account)
REQ-2:The database must not be write protected, ie transactions must be allowed on the server.

REQ-3:The user should have a browser for using the website. (IE 6 or greater is recommended)

3.2 Administrator Account

3.1.1 There is a separate website for the administrator whose main job is to monitor and maintain the website .

3.1.2 Stimulus/Response Sequences

Any form of malpractice(like putting up derogatory content , or false identity issues) when under the administrators notice requires the administrator to take action which may involve a permanent ban on the user's account.

Products being up on the website even after being sold must be removed by the administrator.

2.1.3 Functional Requirements

REQ-1:Should have access to the admin site(i.e have an administrator account)
REQ-2:The database must not be write protected, ie transactions must be allowed on the server.
REQ-3:The user should have a browser for using the website. (IE 6 or greater is recommended

3.3 Intutive and User Friendly GUI

3.3.1

The store operates very intuitively and has a fully functional tested GUI which guides the user to various sections of the site. At all points navigation is made easy by the use of many links to different parts of the website.

3.1.2 Stimulus/Response Sequences

3.1.3 The user uses the website by providing a request to the server. The response involves the rendering of the templates along with the view function.

3.3.2 Functional Requirements

REQ-1:Should have access to the site(i.e Be a BITSian)
REQ-2:The database must not be write protected, ie transactions must be allowed on the server.
REQ-3:The user should have a browser for using the website. (IE 6 or greater is recommended

4. External Interface Requirements

4.1 User Interfaces

The user interface for the software shall be compatible to any browser such as Internet Explorer, Mozilla or Netscape Navigator by which user can access to the system. The user interface can be implemented using any tool or software package like Java Applet, MS Front Page, EJB etc.

Here, we used the help of twitter's bootstrap for implementing the front end. It had defined CSS classes which were used to give the web app a user friendly and simple GUI. The images that are uploaded for products have a constraint of 160*160. Other than that pretty standard interface is there. On the main products page, a loop keeps on adding products through rows. They're also divided into categories through columns.

4.2 Hardware Interfaces

Since the application must run over the internet, all the hardware shall require to connect internet will be hardware interface for the system. As for e.g. Modem, WAN – LAN, Ethernet Cross-Cable.

4.3 Software Interfaces

- The website has the settings.py file which has the rules defining how each module communicates with other related modules.

- The modules in django are supposed to be mentioned. The admin module communicates with the module.
- The python functions communicate with the authentication ,session middleware and django.contrib.auth and the UserApp module to facilitate login.
- The python functions communicate with the common middleware and Template Loaders package to facilitate the rendering of HTML files.
- The python functions communicate with the CSRF (Cross Site Request Forgery) and ***clickjacking*** middleware to facilitate input by users in a safe manner.
- The website communicates with django.contrib.admin to facilitate use of the administrator package.
- The website communicates with the django.db module to store and retrieve items from the database hosted using sqllite3.

- The website communicates with django.contrib.staticfiles package to render static files(CSS, JS and Image files)

4.4 Communications Interfaces

The e-store system shall use the HTTP protocol for communication over the intranet for communication between the user and the server.

5. Other Nonfunctional Requirements

5.1 Performance Requirements

It can handle 100 requests per second maximum as it does not run in a multithreaded environment. As for scalability if hosted on a cloud, performance could scale too. Indexing and caching middleware if implemented could also improve performance significantly. SqlLite 3 runs with only one write operation at a time, switch to MySql for better performance. Varnish implementation to handle non html data could lead to lesser hits coming on to the server.

5.2 Safety Requirements

The loss of personal information leading to privacy issues could occur as personal details are made public. However the redundancy lies in how swd provides the same server explicitly hence we making private information public is okay.

5.3 Security Requirements

User Authentication is done using cookies, session middleware and django.contrib.auth's User model. It uses a one way hash, mostly md5 and for 16 rounds along with salt values to prevent rainbow attacks.

Since there is online payment for now, security concerns are only for personal information being made public.

5.4 Software Quality Attributes

- Because of its c2c product catalogue design, the application is adaptable to sale and purchase of any item that is there for selling and buying. The application allows interaction between the users.

- Since the entire application is independent of operating system, there are no portability problems

- This version (1.0) does not guarantee reliability. But in future versions, authentication of details provided by user by verifying with Student Welfare Division website will be carried out.

- Buying and selling of commodities like books, electronis being common scenario in most campuses, application can be reused in different campuses without any changes.

- Version 1.0 can easily be tested since operations are straightforward. Long term goals requires rigorous testing.
  .

Appendix A: Glossary

- *GUI – Graphical User Interface*

- *HTML – Hyper Text Mark-Up Language*

Appendix B: Analysis Models

Basic Use cases:

Use case: Web customer actor registers with e-store

Description: This use case describes how a new User can register with e-store

P**rimary Actor**: User

Stakeholders and Interests:
- **User**: Wants user-friendly interface and fast searching speed.
  Wants to register and create the account with ease and within a short time.
  Company: Wants to satisfy user interests and validate user information.

Preconditions: E-store website main page is loaded.

Success Guarantee (Post Conditions):
- 'Account verification Screen' lets the user review his/her account details and then successfully register as a user of this site.

Basic flow:
1. The new use click on new 'create new Account link'.
2. The user is at Account Information screen.
3. The new use enters the following details in the Account Information Screen.
   Contact Information:
   a. First Name
   b. Last Name
   c. Profile picture
   d. Hostel Address
   e. Telephone Number
   f. E-Mail ID
4. The user clicks on Update and the system validates all the user information and displays the signing information page.
5. User enters the new Username and Password.
6. System validates that the Username is already in use. If not, system displays the new account confirmation page.

Use case: Web customer actor uses web site to look through the products available online.

Description: This use case describes how the User can search/browse the e-store catalog.

Primary Actor: User

Stakeholders and Interests:
- **User**: Wants user-friendly interface and fast browsing speed.
  Wants to browse the catalog and add items to the cart successfully.
- **Company**: Wants to satisfy user interests.

Preconditions: None

Success Guarantee (Post Conditions):

- 'Product Screen' displays items and corresponding categories for a chosen product.
- Item Screen' displays detailed information about an individual item for sale, including a photo, if one is available.
- Upon clicking a product, the user is shown, in full detail the product's information and the seller's information, like his hostel address, phone number etc.

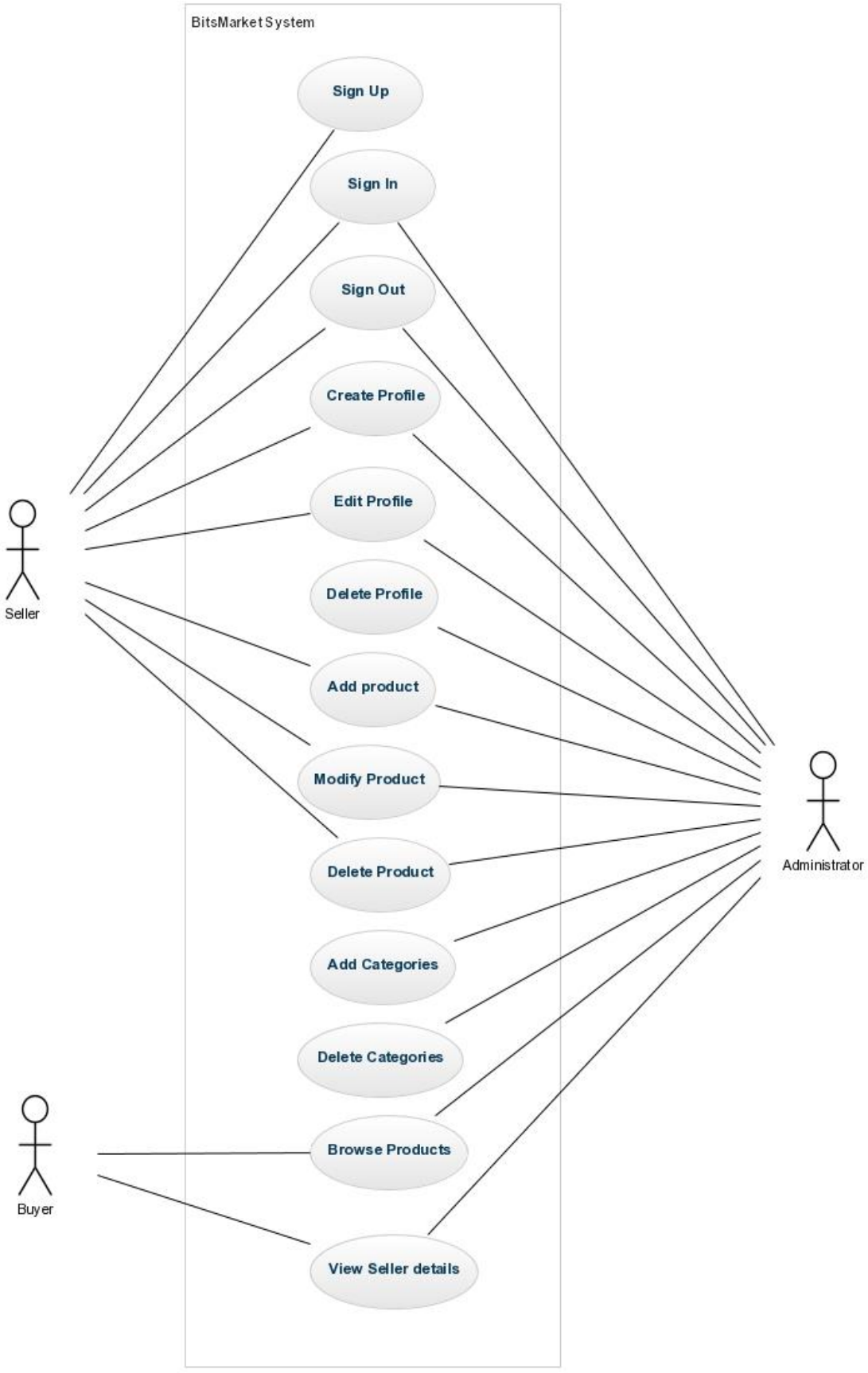
Basic flow:

1. User opens a web browser, gives the URL for the '**BITSMarket'** website in and clicks on 'Go' button.
2. System launches the web site.
3. User clicks on any product link in the 'Product' list, sorted in categories.
4. System displays 'Category Screen' with the products available for the category chosen in Step 3.
5. User clicks on any product link in 'Products for this Category' list.
6. System displays 'Product Screen' with list of all information of the item for the product chosen in Step 5 along with the seller's details.
7. System displays 'Item Screen' for the item chosen in Step 7, including a photo if one is available and seller's full information

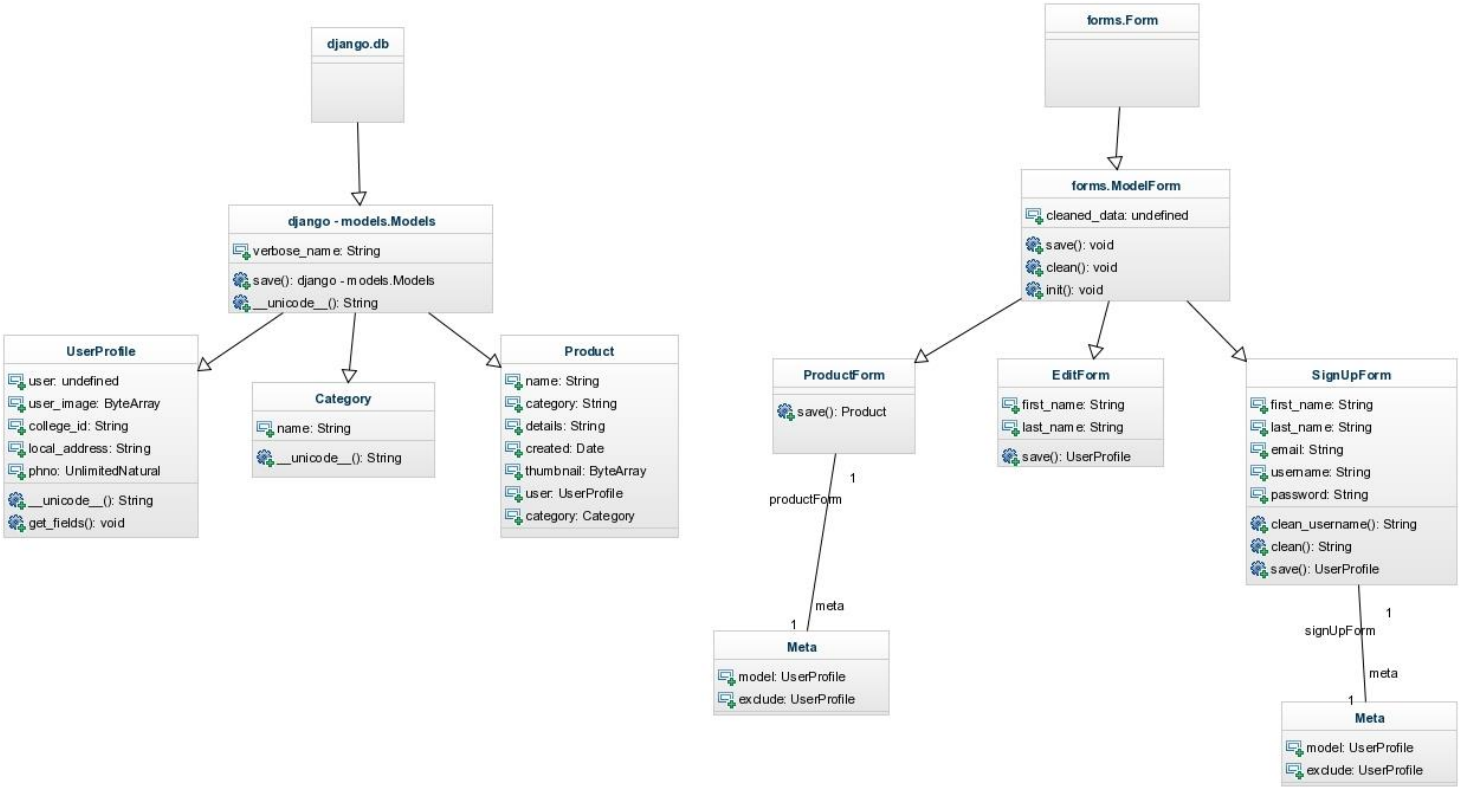
Extensions (Alternate Flow):

3a. User navigates to category page of a particular type of product by clicking on any product in the image map located in the center of the page.

4a. User views the next few items from the list of all products in category by clicking on 'Next' link in the bottom right corner of product list and then navigates to 'Product Screen' of a particular product by clicking on that product link in 'Products for this Category' list.


BitsMarket System
Sign Up
Sign In
Sign Out
Create Profile
Edit Profile
Delete Profile
Add product
Modify Product
Delete Product
Add Categories
Delete Categories
Browse Products
View Seller details
Seller
Buyer
Administrator

UML Diagram:


django.db
forms.Form
django - models.Models
verbose_name: String
save(): django - models.Models
__unicode__(): String
forms.ModelForm
cleaned_data: undefined
save(): void
clean(): void
init(): void
UserProfile
user: undefined
user_image: ByteArray
college_id: String
local_address: String
phno: UnlimitedNatural
__unicode__(): String
get_fields(): void
Category
name: String
__unicode__(): String
Product
name: String
category: String
details: String
created: Date
thumbnail: ByteArray
user: UserProfile
category: Category
ProductForm
save(): Product
EditForm
first_name: String
last_name: String
save(): UserProfile
SignUpForm
first_name: String
last_name: String
email: String
username: String
password: String
clean_username(): String
clean(): String
save(): UserProfile
1
productForm
meta
1
Meta
model: UserProfile
exclude: UserProfile
1
signUpForm
meta
1
Meta
model: UserProfile
exclude: UserProfile