



Press Agency System

Problem Definition:

The news media or news industry are forms of mass media that focus on delivering news to the general public or a target public. These include print media (newspapers, newsmagazines), broadcast news (radio and television), and **more recently the Internet** (online newspapers, news blogs, etc.).

Online journalism is produced or distributed via the Internet, so we need to **develop a news media web application** to collect and distribute online news and blogs to viewers in one web application rather than many distributed offline newspapers.

Actors (Three actors):

- Admin
- Editor
- Viewer

Layouts (Three layouts):

1. We need a web application (**ASP.Net MVC**) which collects all news and blogs in one platform written by different editors in different news fields and view these blogs to the viewers
2. The web application includes 3 layouts
 - First layout: the viewers' layout and we will call it the **wall Layout** (all the news posts exist there)
 - Second layout: The editor layout and we will call it the **factory Layout** (as the editors writes and edits news posts in order to view it to the viewers)
 - Third Layout: The admin layout and we will call it the **dashboard Layout** (where admin can control editors and their posts and also control viewers)

Actors Roles:

1. **Admin:**
 - Control all users (Viewers and Editors): Admin can Add, Remove Users

- Control the **wall** page: Admin can Add, Remove and Update news posts in the wall page which are created by the editors
- Admin pages (in **dashboard layout**) **include**:
 - a. Admin information (profile page): first name, last name, email, phone number, photo, user role (admin)
 - b. All the users that using the system so that the admin can control them.
 - c. All the posts that created by the **editors** so that the admin can control them.
 - d. **If the Admin remove a post from his profile it reflects directly to the wall page and be removed.**
 - e. Admin Accept or Refuse Posts written by the Editors before adding it to the **Wall**.
 - f. The admin must login first, to operate his job

2. Editor:

- Write news posts that will show in the **wall** page
- Editor pages (in **factory layout**) **include**:
 - a) Editor information (profile): user name, first name, last name, email, phone number, photo, user role (Editor)
 - b) Editor can change (update) his information except his role
 - c) Password **Not Showed** but we need a button to allow Editor to Change his password
 - d) A Form that the Editor will use to write news posts and add them to the Wall.
 - e) **News Post** contains: Editor Name, Article Title, Article body, Post Creation Date, Article type (Sports, Cinema, etc...), Number of viewers, Article Image.
 - f) Admin Accept or Refuse Posts written by the editor before adding it to the **Wall**.
 - g) History that includes all news Posts (Articles) that have been added before by this editor in the Wall (mention date in each Post)
 - h) Editor can manage his Posts (Add, Edit, Delete, Search).
 - i) Editor can answer viewers' questions.
 - j) The Editor must login first, to operate his job

3. Viewer

- Login to interact with news articles

- View (read only) all news posts from different editors (no need to login)
- Viewer pages (in **wall layout**) **include**:
 - a) All Posts that have been added by Editor from their Profiles
 - k) Each post (created by Editor) has: Editor Name, Article Title, Article body, Post Creation Date, Article type (Sports, Cinema, etc...), Number of viewers, Article Image.
 - b) Viewer can search for a job by Title, Editor name, Article type.
 - c) Viewer can save a particular post in saved page (to read later)
 - d) Viewer can open each article in separate page to see all the article body
 - e) Viewer can like or dislike the Article (number of likes and dislikes will show in each article)
 - f) Viewer can Ask Questions and Editors will answer (FAQ).

Important Notes:

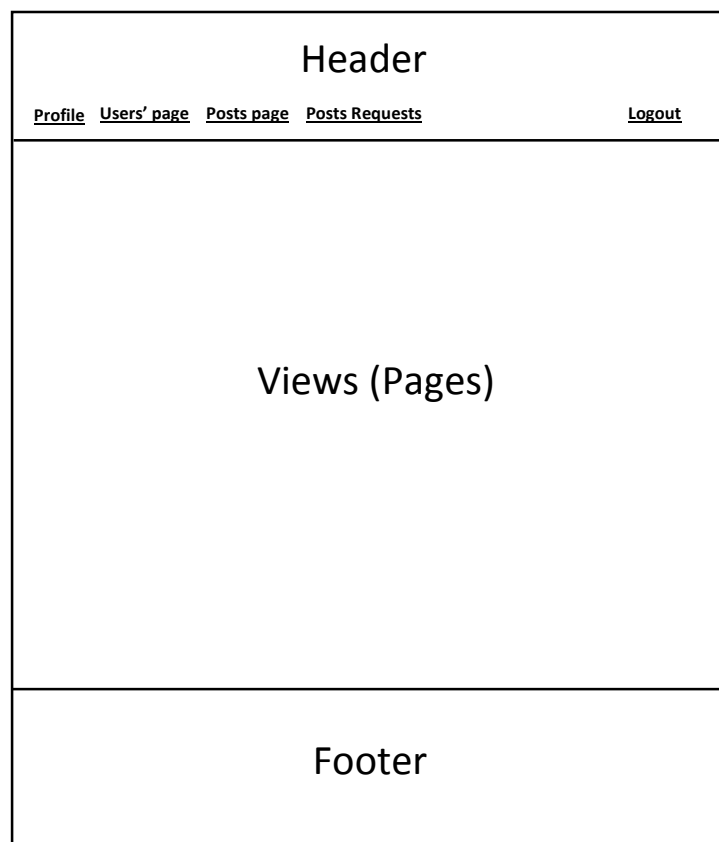
- You need to think and **design your project relational database** (tables, tables' columns and tables' relationships) and **Print the Database Schema**
- You have three layouts (Wall – Factory - Dashboard) and Many Views (Pages).
- Views Folder should include:
Shared folder contains (WallLayout.cshtml, FactoryLayout.cshtml, DashboardLayout.cshtml)
- Create two Popups (modal):
 - a. First Popup: For Login
 - b. Second Popup: For Registration.
 - The Login popup appears when a button in the Wall layout is clicked.
 - The Registration popup appears when a button in the Wall layout is clicked.
 - If the viewer or the editor doesn't have an account then he will register.
 - Registration Information: first name, last name, username, email, phone number, photo, User role (viewer or editor). **The role can't be updated later**
 - You can add admin in the database manually and then you can login.
- Your app is responsive (**Bootstrap is required for Grid system**)

Layouts:

1- Dashboard Layout (Admin)

- Include these pages (as mentioned above):
 - a. Admin profile page
 - b. Users page (add or delete users (viewers and editors))
 - c. Posts page (delete or update posts which is already in the wall)
 - d. Posts requests (accept or refuse to add new posts (created by the editors) in the wall)

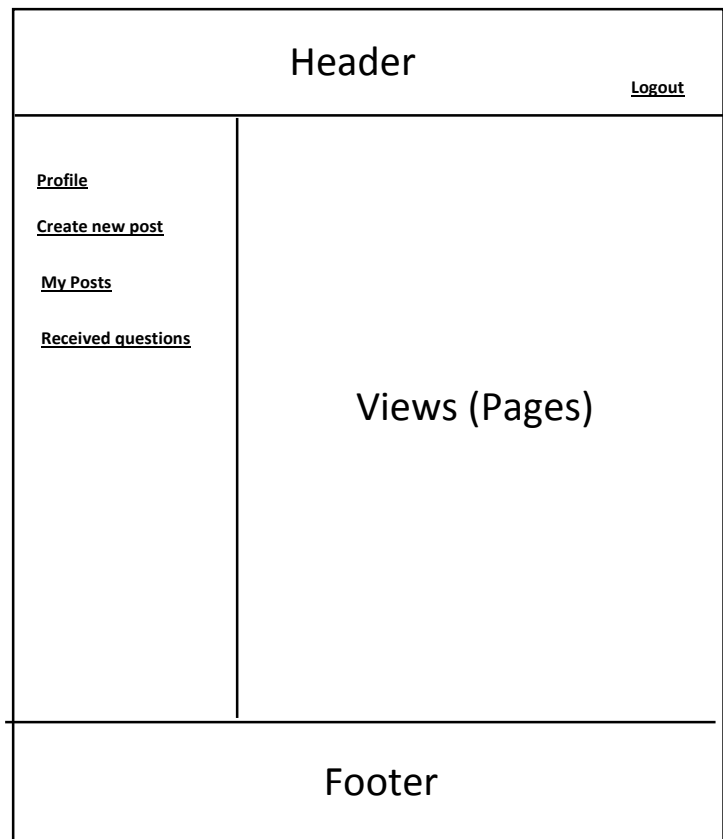
Dashboard Layout:



2- Factory Layout (Editor)

- Include these pages (as mentioned above):
 - a. Editor profile page (profile information is changeable except the role)
 - b. Create news post page (to write news posts)
 - c. History page (contains posts created by this editor and can delete and edit and search these posts)
 - d. Received questions (answer questions asked by the viewers for a certain article)

Factory Layout:



3- Wall Layout (Viewer)

- Include these pages (as mentioned above):
 - a. The wall page: which contains all the news posts
You can view this page without logging (read only). Viewer functionality (as mentioned above) will be available after logging in.
 - b. The save page: viewer saves job posts in it to read it later.
 - c. Viewer can search for a job by Title, Editor name or Article type (**search in the layout**)
 - d. Login and register forms are popups

Wall Layout:

Header	
Filter: <input type="text"/>	Register Login
Footer	

Best Wishes