Structural Model.

Structural model is a UML diagram that represents the static behaviours like relationship and properties. Class diagram, Flowchart and DFD are theexamples of a Structural model. The decription of class diagram and flowchart is given below.

Class Diagram.

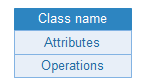
Introduction.

Class diagram is a static diagram. It represents the static view of an application. Class diagram is not only used for visualizing, describing, and documenting different aspects of a system but also for constructing executable code of the software application.

Justification.

Class diagram is used to show the static modelling of my system. It also describes the responsibilities of a system. It shows the collaboration among the elements of static model.

Notation use.



Class: Class shows the architecture and features of designed system.



Aggregation: Aggregation describes group of object and how you interact with them.



Dependency: Dependency shows that one element is dependent on other. For e.g. client depends on the supplier.



Composition: It represents whole-part relationships and is a form of aggregation.



Generalization: In this relationship one element is based on another element.



Association: is a relationship between two classifiers, such as classes or use cases, that describes the reasons for the relationship and the rules that govern the relationship.

Diagram.

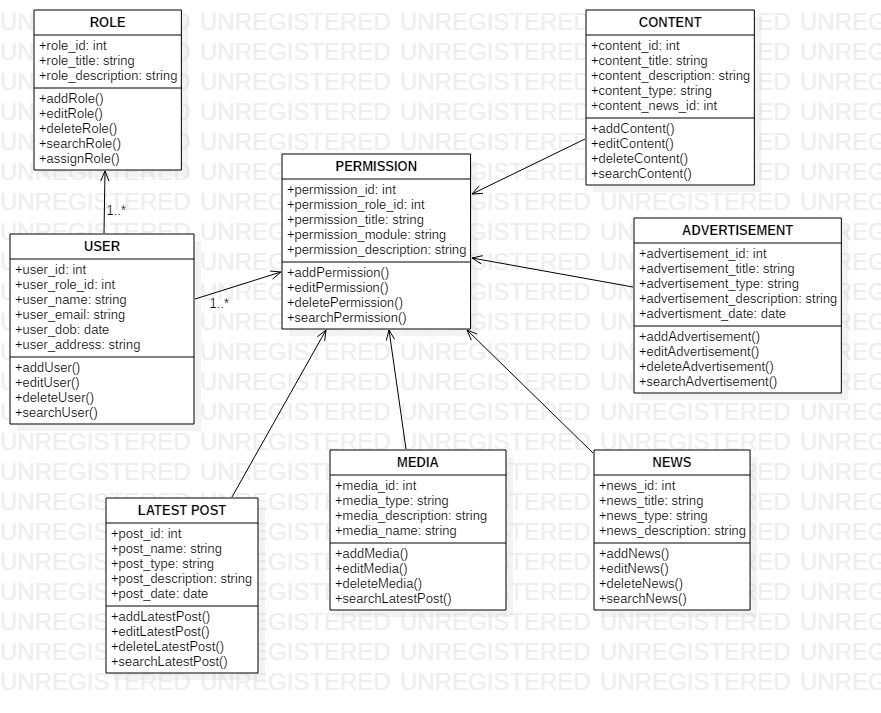


Figure : Final class diagram

Explanation.

News portal system class diagram describes about the news portal classes, their attributes, operations (or methods), and the relationships among object. The main classes of the News Portal System are News, Contents, Media, Advertisement, Latest posts, News Type.

Classes of News Portal System Class Diagram:

* News Class: Manage all the operations of News.
* Contents Class: Manage all the operations of content.
* Media Class: Manage all the operations of Media.
* Advertisement Class: Manage all the operations of Advertisement.
* Latest Posts Class: Manage all the operations of Latest posts.
* News Type Class: Manage all the operations of News Type Class.

Flow-Chart.

Introduction.

A flowchart is a diagram that shows the process, system or algorithm. Flowcharts are used in multiple field to document, study, plan, improve and communicate complex process.  They can range from simple, hand-drawn charts to comprehensive computer-drawn diagrams depicting multiple steps and routes.

Justification.

A flowchart id drawn to visually represent the processes of a system. It helps to identify the steps and materials needed of the process. There are different problems that may occur in a project but the flowchart helps to clarify the logic of a problem. It also helps in the coding part. It is a quick method of showing program flow.

Notation Use.

Start/End: Oval Shape represents the start or end point of a process.

Process: Rectangle box represents process which shows the processing of action.

Input/Output: this Parallelogram shape is used to either show the input or output operation.

Flow-line: It shows the direction of logic flow in a program.

Decision: denotes a decision made to be yes or no. Program should continue along one of the routes.

Diagram.

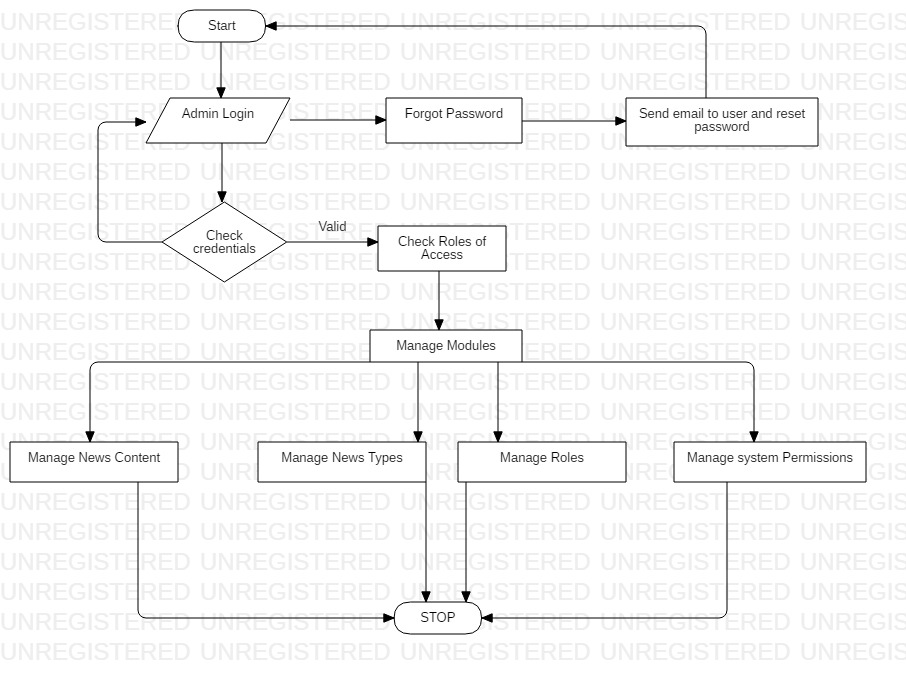


Figure : Flowchart

Explanation.

Above flowchart shows the flow of logic from the diagram. Here the admin logs in the system, the system checks necessary credentials. If the given information is right then system provides further access to admin, if not then an alert message is shown saying ” given information is wrong” and returns admin back to the login page. If the admin forgets his password, then he can reset the password using forget password option. Once admin accesses the system, he can manage news, categories, contents advertisements etc.

Behavioral model.

Behavioral models are models of the dynamic behavior of the system as it is executing. They show what happens or what is supposed to happen when a system responds to a stimulus from its environment. Examples of behavioral model are activity diagram, sequence diagram.

Sequence Diagram.

Introduction.

Sequence Diagrams are interaction diagrams that detail how operations are carried out. They capture the interaction between objects in the context of a collaboration. Sequence Diagrams are time focus and they show the order of the interaction visually by using the vertical axis of the diagram to represent time what messages are sent and when.

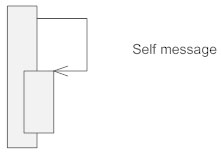
Justification.

While there is the assumption that [Sequence Diagrams](https://creately.com/Draw-UML-and-Class-Diagrams-Online) were made for developers, the truth is that a project workers could use such diagrams to communicate how exactly the project presently currently works by illustrating how the different project objects interact. This allows the specification of simple runtime scenarios in a graphical manner.

Notations use.

Lifeline: lifeline notation with an actor element symbol is used when the particular sequence diagram is owned by a use case.

Actor: A lifeline with an entity element represents system data. For example, in a customer service application, the Customer entity would manage all data related to a customer.



Self-message: it is the message that an object sends to itself.



Found-message: A message sent from an unknown recipient, shown by an arrow from an endpoint to a lifeline.



Synchronous: A synchronous message requires a response before the interaction can continue.

Diagram.

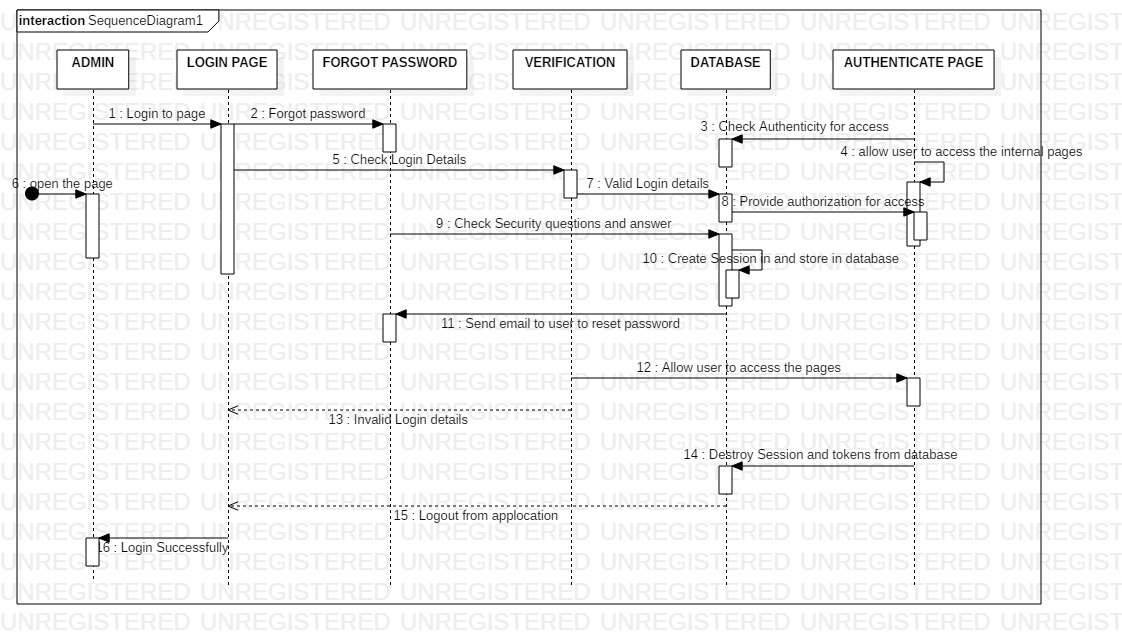


Figure :Sequence Diagram for admin Login

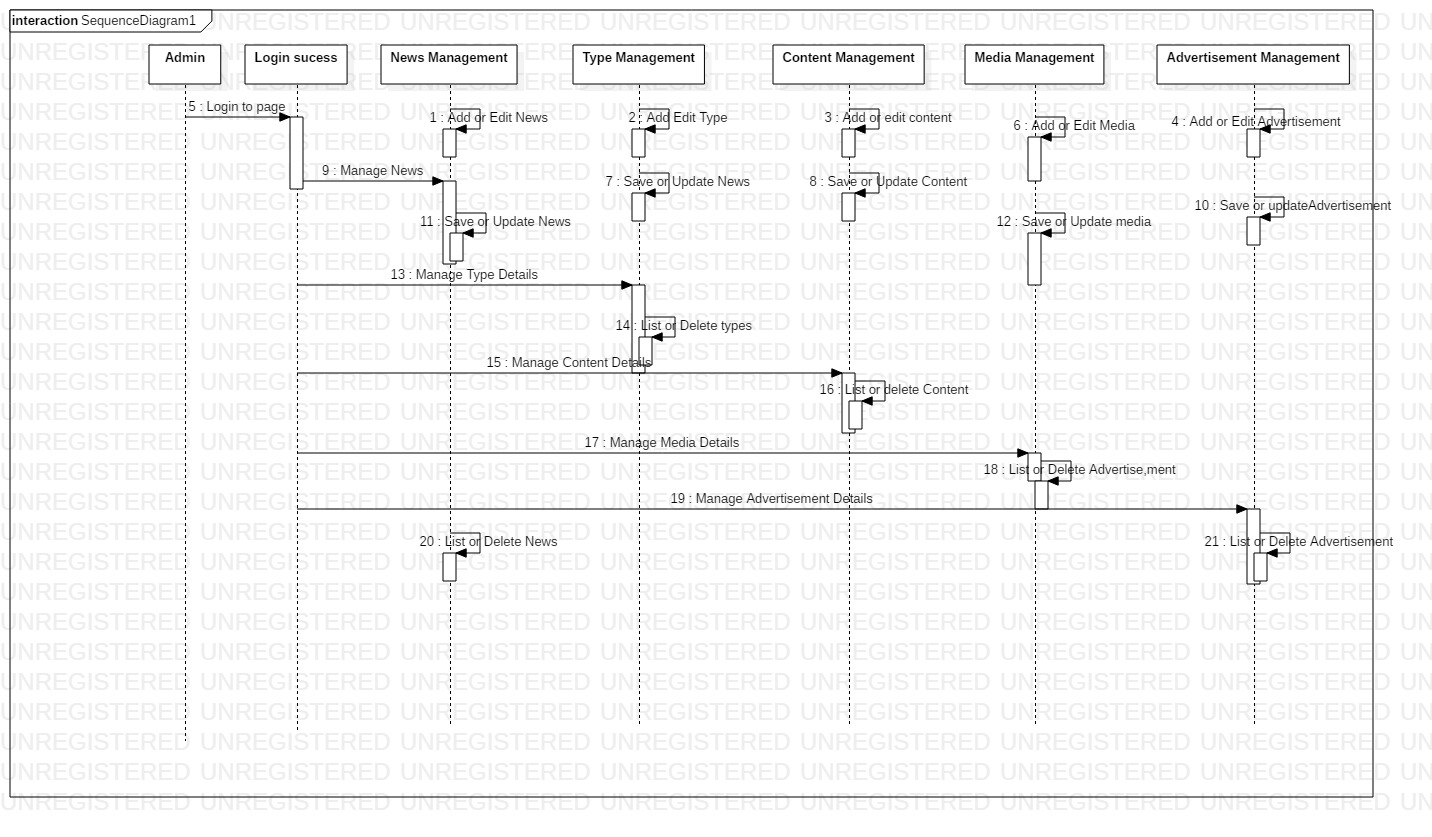


Figure :Sequence diagram of whole System.

Explanation.

Above is the Sequence diagram for News Portal System that shows the interaction between the objects of News, Types, Contents, Latest Posts, Media. The instance of class object involved in this diagram are News Object, types Object, Contents Object, Latest posts Object, media Object.

Activity Diagram.

Introduction.

Activity diagram is basically a flowchart to represent the flow from one activity to another activity. The activity can be described as an operation of the system. Activity diagrams deal with all type of flow control by using different elements such as fork, join, etc. It captures the dynamic behavior of the system.

Justification.

Activity diagram is used to describe a function of the system represented by a use case diagram. It also models the activities that make up the life cycle in the unified process. It also models a task for e.g. in business modelling. The basic purpose why activity diagram is made are similar to other diagrams i.e. to capture the dynamic behavior of the system.

Notation Use.

Initial Node: A small filled circle followed by an arrow represents the initial action state or the start point for any activity diagram.

Action Flow: Action flows, also called edges and paths, illustrate the transitions from one action state to another. They are usually drawn with an arrowed line.

Activity: An action state represents the non-interruptible action of objects.

Decision: this diamond shape must be true before going to next step. These are not essential, but are useful when a specific answer.



Joint: Combines two concurrent activities and re-introduces them to a flow where only one activity occurs at a time.



Fork: Splits a single activity flow into two concurrent activities.

Diagram.

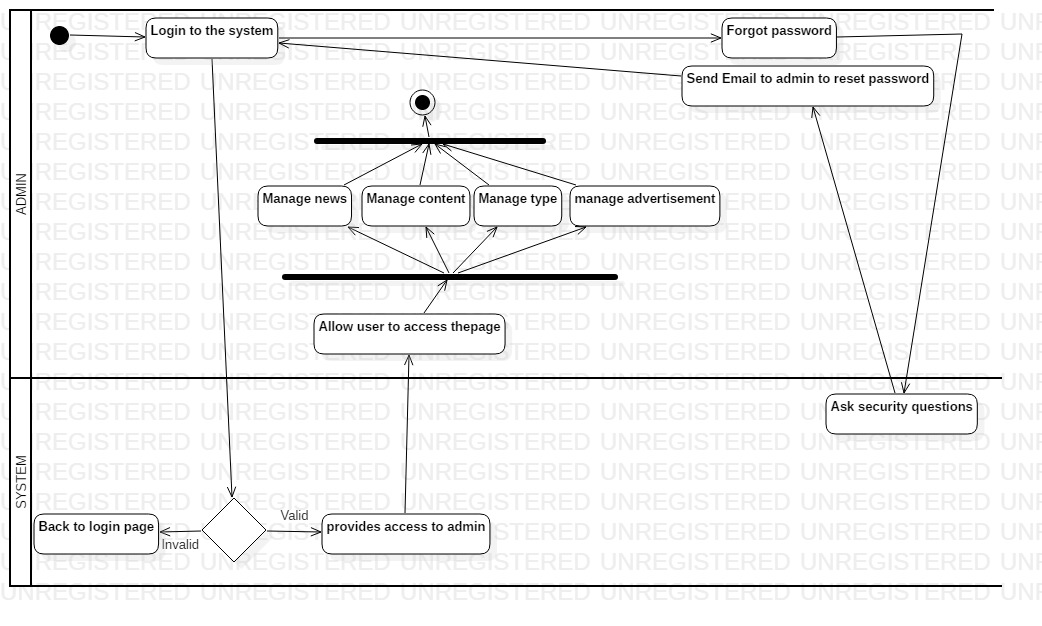


Figure : Activity diagram.

Explanation.

Firstly admin logs in to the system, then the system checks the necessary credentials and if the credentials are right then, it provides access to the admin. After the admin accesses the system, admin can manage news, manage content, manage types, manage advertisement etc. If in case the admin forgets the password, then it asks the security questions and email is sent to admin to reset the password and then he can login to the system and process the further things.

Database.

ER Diagram.

Introduction.

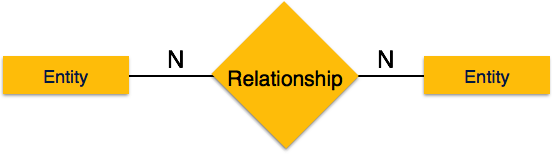
An entity relationship diagram (ERD), also known as an entity relationship model, is a graphical representation of an information system that depicts the relationships among people, objects, places, concepts or events within that system. Entity relationship diagrams provide a visual starting point for database design that can also be used to help determine information system requirements throughout an organization.

Justification.

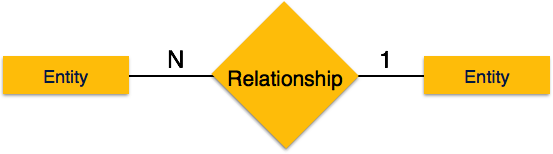
Creating an ERD for a proposed database design allows us to see a high-level view of our database before we actually get to creating it, allowing us to make sure we’re capturing everything we need to capture in our various tables and make tweaks before we actually create the database (when it will be much more difficult to make changes).

Notation Use.

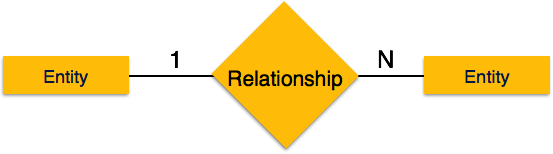
Entity: Any real-world object can be represented as an entity about which data can be stored in a database. All the real world objects like a book, an organization, a product, a car, a person are the examples of an entity.



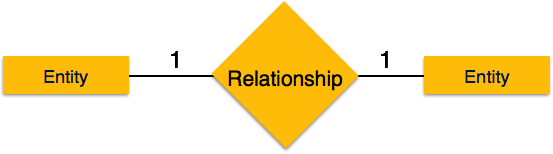
Many-to-Many: The above image reflects that more than one instance of an entity on the left and more than one instance of an entity on the right can be associated with the relationship. It depicts many-to-many relationship.



Many-to-one: The above image reflects that more than one instance of an entity on the left and only one instance of an entity on the right can be associated with the relationship. It depicts many-to-one relationship.



One-to-Many:  The above image reflects that only one instance of entity on the left and more than one instance of an entity on the right can be associated with the relationship. It depicts one-to-many relationship.



One-to-One: The above image reflects that only one instance of each entity should be associated with the relationship. It depicts one-to-one relationship.

Diagram.

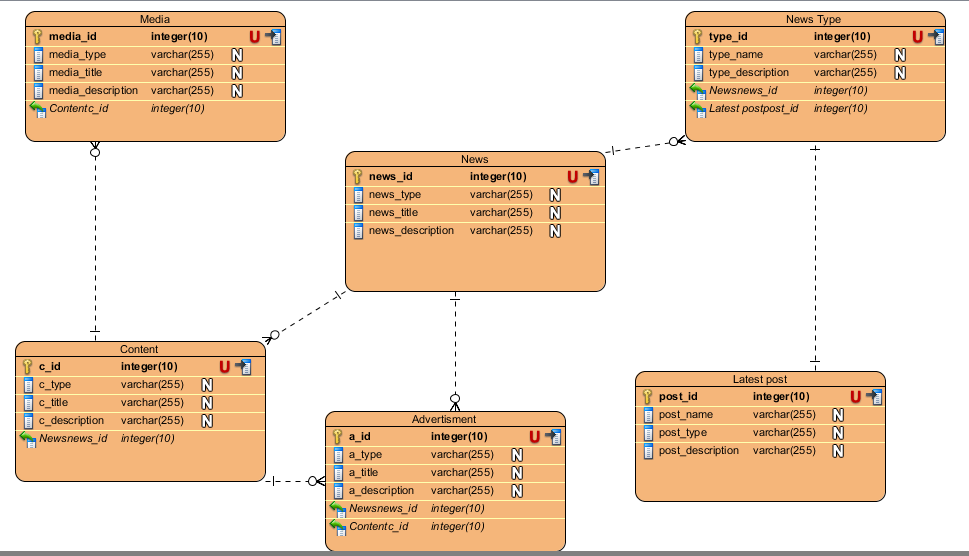


Figure :ER diagram

Explanation.

Each entity (News type, Media, latest post, advertisement) contains primary key and are unique. The entity media, latest post has binded with news, content entity with foreign key. There is one-to-one and one-to-many relationships available between latest posts, advertisement, news type and news. All the entity are normalized and reduced duplicity of records. I have implemented indexing on each tables of News portal system for fast query execution.

Data Dictionary.

Introduction.

A data dictionary is a file or a set of files that contains a database's metadata. The data dictionary contains records about other objects in the database, such as data ownership, data relationships to other objects, and other data. The data dictionary is a crucial component of any relational database. Ironically, because of its importance, it is invisible to most database users. Typically, only database administrators interact with the data dictionary.

Justification.

First and foremost, in the mind of many manufacturers will be the cost associated with prototyping a product. Likewise, digital prototyping will help to ensure that this stage of production is as short as possible. Since digital prototyping can be conducted more quickly, it follows that the prototyping stage of production can be a great deal more thorough.

Diagram.

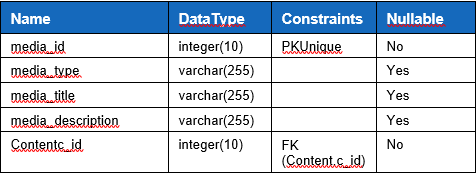


Figure :data dictionary for media

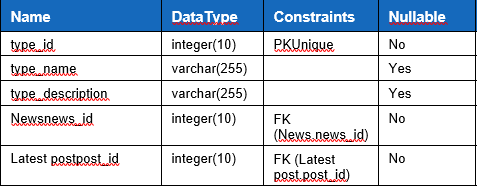


Figure :data dictionary for news type

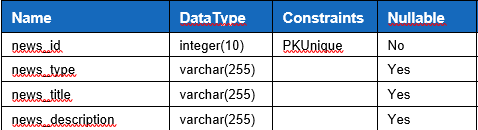


Figure :data dictionary for news

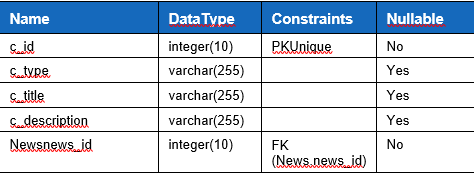


Figure : data dictionary for content

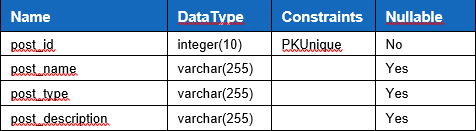


Figure : data dictionary for latest post

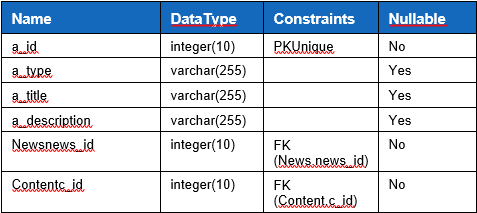


Figure : data dictionary for advertisement

UI

Prototype.

Digital Prototyping goes beyond simply creating product designs in 3D. It gives product development teams a way to assess the operation of moving parts, to determine whether or not the product will fail, and see how the various product components interact with subsystems either pneumatic or electric. Digital Prototyping changes the traditional product development cycle from design>build>test>fix to design>analyze>test>build. Instead of needing to build multiple physical prototypes and then testing them to see if they’ll work, companies can conduct testing digitally throughout the process by using Digital Prototyping, reducing the number of physical prototypes needed to validate the design.

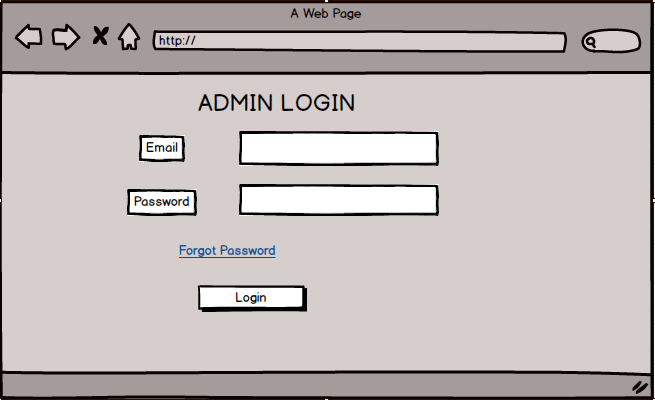


Figure : Admin Login.

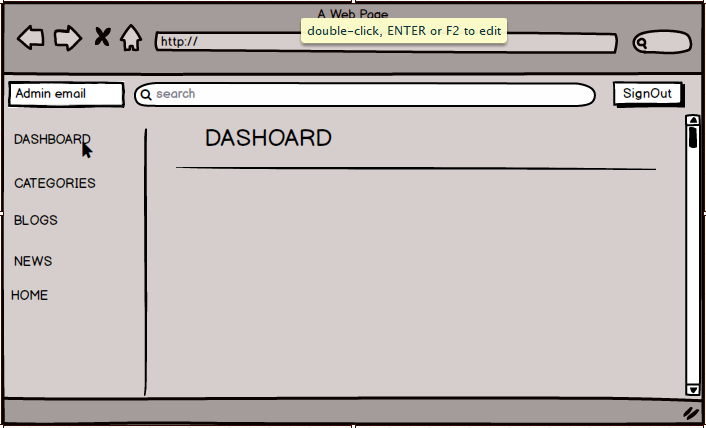


Figure :Dashboard.

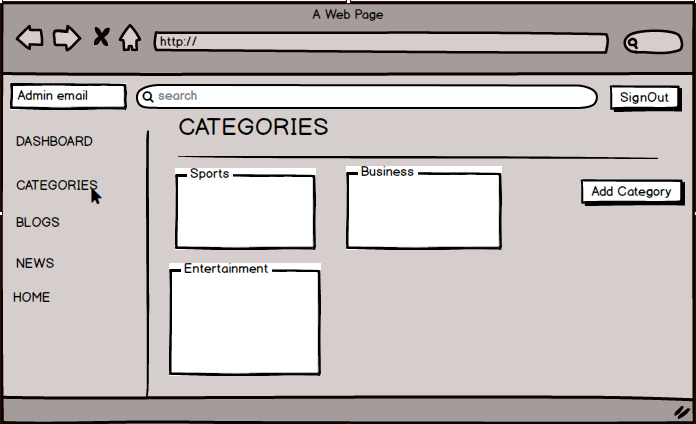


Figure :Categories section.

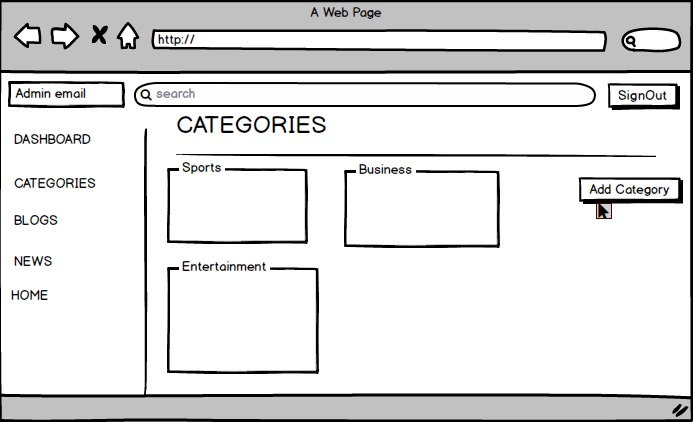


Figure :Add category button

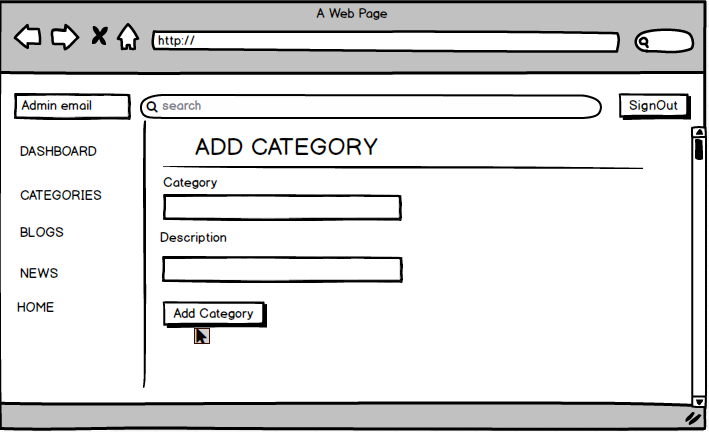


Figure :Adding category.

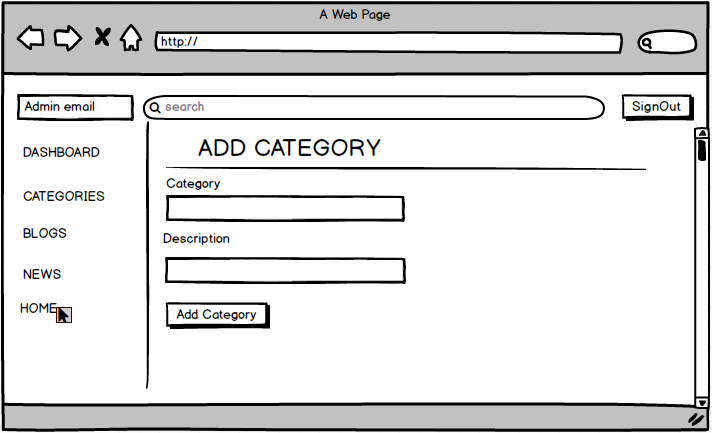


Figure :Category section.

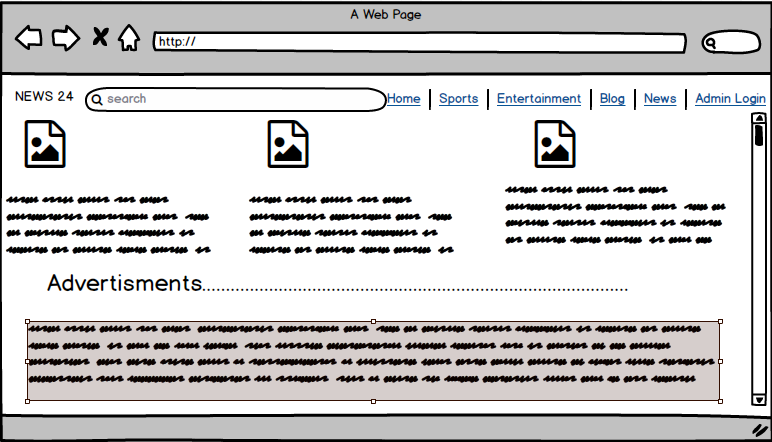


Figure :Home page

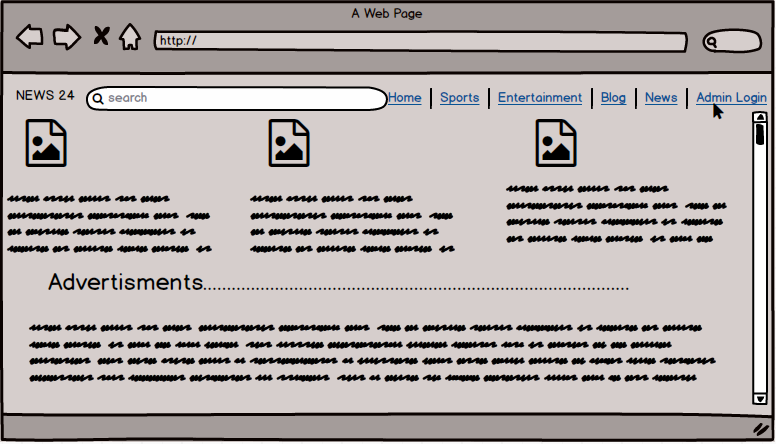


Figure :accessing admin page

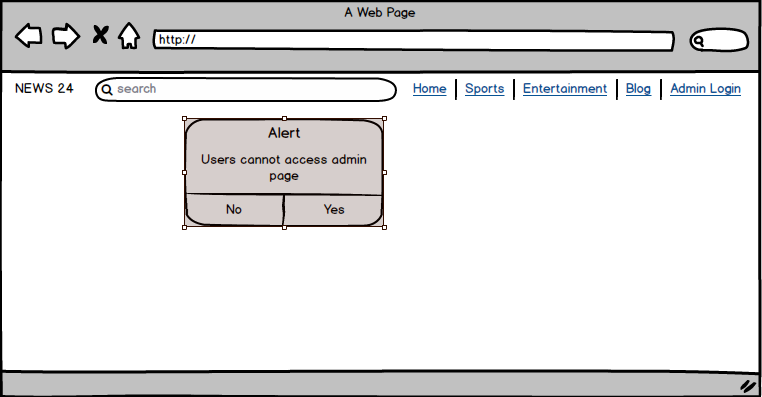


Figure : Alert Message