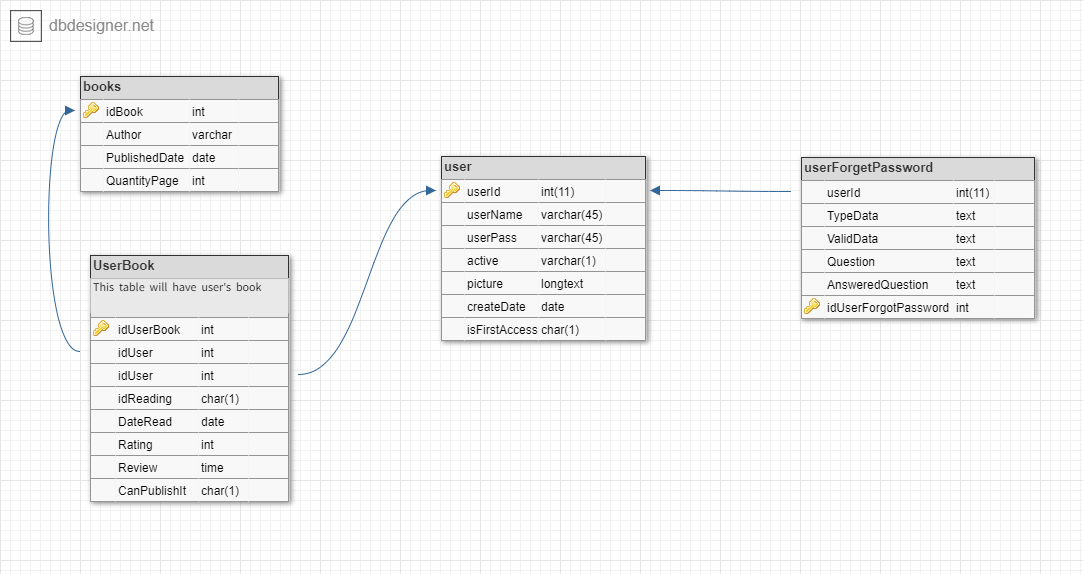
**System Requirements**

|  |  |
| --- | --- |
| dETAILS of project | |
| PROJECT name | MyReviewBook |
| Description | Users will write their own reviews and rating and they could share their opinion on the main screen. |
| Technologies envolved | * C# (EntityFramework) * Javascript * CSS * Bootstrap * MySql |
| resources to design | Data base: [app.dbdesigner.net](https://app.dbdesigner.net/)  Mockups: <app.moqups.com> |
| Hosting | |  |  | | --- | --- | | Details | | | Database  Google cloud | Configuration   * MySQL version: 5.7 * vCPU: 1 * HD SSD: 10GB * Memory RAM: 3,75GB * Public IP: 35.188.222.189 * Connexion Instance: meta-yen-272217:us-central1:myreviewbooks | | System  Google cloud | Configuration   * vCPU: 2 * HD SSD: 50GB * SO: Linux * Memory RAM: 3,75 * Public: - | |
|  |  |

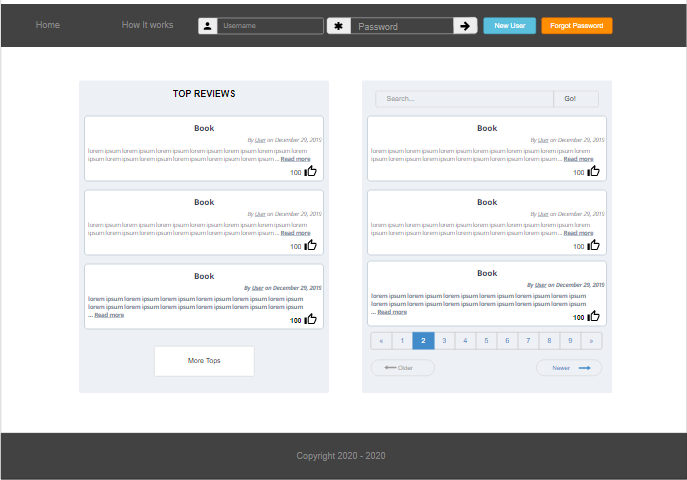
# **Data Base**

* 1. Modeling
  2. Scripts

|  |  |
| --- | --- |
| mYREVIEWBOOK | |
| dATABASE | CREATE DATABASE `myreviewbook`; |
| TABLE USER | CREATE TABLE `user` (  `userId` int(11) NOT NULL AUTO\_INCREMENT,  `userName` varchar(45) NOT NULL UNIQUE,  `userPass` varchar(45) NOT NULL,  `active` varchar(1) NOT NULL,  `picture` longtext NOT NULL,  `createDate` DATE NOT NULL,  `isFirstAccess` char(1) NOT NULL,  PRIMARY KEY (`userId`)  ); |
| TABLE user Forget Password | CREATE TABLE `userForgetPassword` (  `idUserForgotPassword` int NOT NULL AUTO\_INCREMENT,  `userId` int(11) NOT NULL,  `TypeData` TEXT NOT NULL,  `ValidData` TEXT NOT NULL,  `Question` TEXT NOT NULL,  `AnsweredQuestion` TEXT NOT NULL,  PRIMARY KEY (`idUserForgotPassword`)  ); |
| TABLE BOOKS | CREATE TABLE `books` (  `idBook` int(11) NOT NULL AUTO\_INCREMENT,  `Author` TEXT,  `PublishedDate` DATE,  `QuantityPage` int,  PRIMARY KEY (`idBook`)  ); |
| TABLE USER BOOK | CREATE TABLE `UserBook` (  `idUserBook` int NOT NULL AUTO\_INCREMENT,  `idUser` int NOT NULL,  `idReading` char(1),  `DateRead` DATE,  `Rating` int,  `Review` TIME,  `CanPublishIt` char(1) NOT NULL,  PRIMARY KEY (`idUserBook`)  ); |
| CONSTRAINTS | ALTER TABLE `userForgetPassword` ADD CONSTRAINT `userForgetPassword\_fk0` FOREIGN KEY (`userId`) REFERENCES `user`(`userId`);  ALTER TABLE `UserBook` ADD CONSTRAINT `UserBook\_fk0` FOREIGN KEY (`idUser`) REFERENCES `books`(`idBook`);  ALTER TABLE `UserBook` ADD CONSTRAINT `UserBook\_fk1` FOREIGN KEY (`idUser`) REFERENCES `user`(`userId`); |
| API Google book | URL: <https://www.googleapis.com/books/v1/volumes>  Parameters: ?q=title=VARIABLE  JSON example:    It was used a program that convert a JSON file into a Class C# called “jsonclassgenerator”.  Ref.: <https://archive.codeplex.com/?p=jsonclassgenerator> |

# **Screen (Prototype)**

# Index Page



The Index page will have one navbar with the fields to login (*UserName* and *Password*), two buttons "*New User*" and other "*Forgot Password*" and one link to "How it works", and on div as a footer with a Copyright

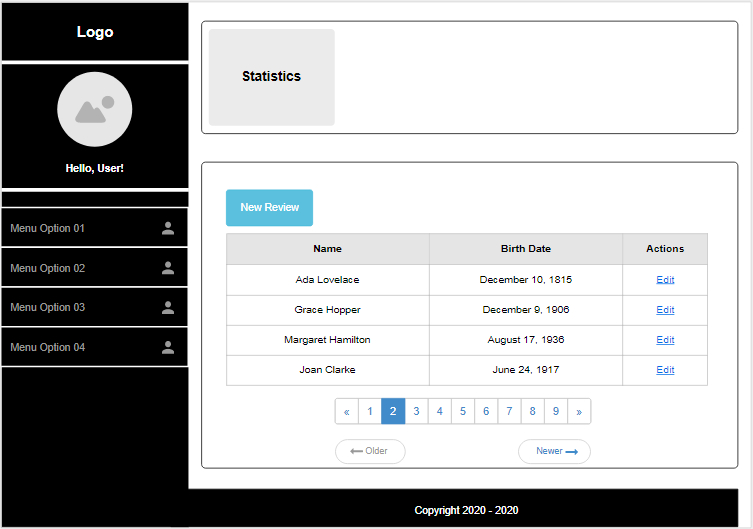
On the center will have the following div:

* Top reviews wrote by users, like a card and any users will be able to give a like
* Any visitant could search to reviews

Page "How it works" is about what are the main goals and how use the site.

Page “Forgot Password” is a three pages steps:

* Step 01: User should type his ***userName***.
  + **If it’s correct**: it goes to Step 02.
  + **If not correct**: show a message error!
* Step 02: User should type his correct ***data previous registered***
  + **If it’s correct**: it goes to Step 03.
  + **If not correct**: show a message error!
* Step 03: User should type his ***answer for a question*** ***previous registered***
  + **If it’s correct**: it shows the **Password**.
  + **If not correct**: show a message error!
  1. Intern Page



After the user logged if it is the first-time user should give the information below, in case he forgets his password.

* Step 01: User should type ***one data like Birthdate, Phone etc***
  + it goes to Step 02.
* Step 02: User should type ***an answer and a question***
  + it goes to Intern page.

The Intern page will have one sidebar a Logo, a default picture, Hello User and the following options

* **Profile**:
  + Change Password.
  + Activate or Deactivate account, it means that the user could just do the login.
  + Logoff
* **Book**:
  + New review
  + Search a book

# **Business’ rule**

* 1. Module user

|  |  |
| --- | --- |
| Index Page | The user can register himself giving one userName and Password |
| To login the user has to provide userName and Password |
| The system has to validate them   * If it doesn't correct, return error * If correct redirect to Intern Screen |
| In case the user forget password   * If the user forgot his password the system will provide three steps for it * First step: The user needs to type his User * Second step: The user needs to provide a valid data * Third step: To ask correctly one question |
| Intern Page | For to the first login   * The user has to follow two steps in case he forgot the Password   + First step: The user has provided on date about him   + Second step: The user has created one question and answer about anything |
| In the Intern Page the user should be able to:   * Change his Picture * Change his Password * Active or Deactivate his account * Logoff |
| Intern Page  (Administration) | This option only will show to user “admin” and it will be composted by:   * Configurate Service to bring data on Google WebService |

* 1. Module Book

The user will be able to register a new review giving the following data NameBook, Author, Published Date, Quantity Page, is Reading, Date Read, Rating, Review, can Publish It.

The system will provide the data (NameBook, Author, PublishDate, QuantityPage) in case someone has already registered it.

The user will be able to search registered books.

In the both way the system will look the book in own database, if the system doesn't find it the system will call a webservice to try to find it.

It will have two scenarios:

First: The user had read the book then the following fields will be mandatory such as Name Book, Rating, Date Read, Review and, if the user would like to share it.

Second: The user will read or he is reading then the following fields will be mandatory such as Name Book and is Reading.

* 1. Module Service (API Google)

This Service will be charge to search at WebService the following data as Name Book, Author, Publish Date, Quantity Page

* 1. Module Administrator

On the system will have a specifics options about Settings, these options will allow the administrator set some configurations about the API related with this project.

URL API Rest, User and Password (if it has), TAGs like name of book, author, publish data, number of pages.

# **Sprints**

* 1. Sprint 01

Layout's design index and intern

* Index Page with:
  + Login
  + Register new Users
  + Login's validation
  + Intern Page with:
  + Hi User
  + Change password
  + Logoff

BUGs

* Name: Timeout Session User
* Problem: The user can go back to intern page without login, but the system loses the session.
  1. Sprint 02

Module Users:

* Active or Deactivate user
* When user is deactivated, they just do a Login, no more.
* Change picture (They should set up an URL, if not the system set up the default image)
* First Access to register some data to recover the password
  + Creation table
  + Valid Date:
    - Birthday, email, phone, mother’s name, father’s name
  + Question and answer
    - Anyone that user had chosen
* Button “Forgot password” in the Index Page
  + Create three steps at the Forgot Page to recover the password
    - Step 01: Type user and do a validation
    - Step 02: Type the valid data
    - Step 03: Type the correct answer

Fixed Bug

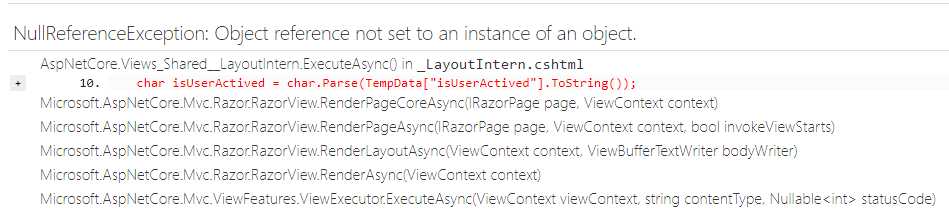
* **Name**: Timeout Session User

**Problem**: The user can go back to intern page without login, but the system loses the session and this mess up with database.

**Solution**: On DashboardControl was set a new contidicton, if userSession == null the user will be redirect to the index page

BUGs

* **Name**: Error at variable **isUserActived**
* **Problem**: Error Layout Intern a variable **isUserActived** when user access other page. Error about reference.
* **Evidence**:



* 1. Sprint 03

Module Books:

* Created table related to books and their constraints
  + Books, UserBook
* Created catalog’s view.
  + The user can type a title of the book that he wants to look for.
  + The system will search it on in own database
  + The system will search it on the Google’s Book API
* The system will provide a list of books
  + The system will show what books already have reviews and what doesn’t have.
* Created manager’s views book
  + The system will provide the data’s book
  + The user can go directly and create a new book filling all fields.
    - The system will provide a list based on the title’s book that the user had given.

Fixed Bug

* **Name**: Error at variable **isUserActived**
* **Problem**: Error Layout Intern a variable **isUserActived** when user access other page. Error about reference.
* **Solution**: All code responsible for Login was in UserController, now it is on Dashboard Controller and it was refactored to be as much abstract as possible that way I can use these methods anywhere in the project.