

RĪGAS VALSTS TEHNIKUMS

DATORIKAS NODAĻA

Izglītības programma: *Programmēšana*

PASKAIDROJOŠAIS RAKSTS

Tēma: “Digitālo grāmatu bibliotēka”

Izpildīja: Mārtiņš Madalāns DP3-3

Rīga 2024

SATURS

IEVADS	3
1. UZDEVUMA NOSTĀDNE.....	4
1.1 Sistēmas lietotāji un to mijiedarbība ar sistēmu	4
1.2 Uzdevumu risināšanas līdzekļu izvēles pamatojums	5
2. DATU MODELĒŠANA.....	6
2.1. Objektu orientēts konceptuālais datu modelis.	6
2.2 Entītiņu relāciju datu modelis.....	7
3. DATU BĀZES IZSTRĀDE	9
3.1 Datu bāzes fiziskā struktūra	9
3.2 Datu bāzes optimizēšana.....	12
4. SISTĒMAS FUNKCIONĀLO PRASĪBU REALIZĒŠANA	13
4.1. Datu pievienošana, labošana, dzēšana	13
4.2. Datu validācija	13
4.3. Datu izgūšana un sapstrāde.....	14
4.4. Datu integritātes nodrošināšanas metožu izmantošana.....	15
4.4.1. Lietotāju autentifikācija	15
4.4.2. Funkcionālo pienākumu sadalīšana.....	15
NOBEIGUMS	16
INFORMĀCIJAS AVOTI.....	17
PIELIKUMI.....	18
Datu bāzes ER modelis	18
Datu bāzes tabulu shēma.....	19
Reģistrācijas pirmais.....	20
Autentifikācijas pirmais	21

IEVADS

Mērķis ir izveidot digitālo grāmatu bibliotēkas mājaslapu, kas piedāvās plašu pašizaugsmes un personīgās attīstības grāmatu klāstu angļu valodā. Mājaslapa kalpos kā resurss ikvienam, kas vēlas uzlabot savas prasmes un zināšanas biznesā, produktivitātē, veselībā un citās jomās.

Šis projekts ir domāts plašai auditorijai, ietverot dažādus cilvēku slāņus ar dažādām interesēm un mērķiem attiecībā uz personīgo attīstību. Galvenie mērķauditorijas segmenti ir:

- Pieaugušie, kas vēlas ieviest pozitīvas pārmaiņas savā dzīvē.
- Karjeras profesionāļi, kas vēlas uzlabot savas prasmes un zināšanas.
- Studenti, kas meklē resursus personīgai izaugsmei.
- Ikviens, kas vēlas paplašināt savu redzesloku un iegūt jaunas zināšanas.

Šis projekts ir aktuāls, jo pašizaugsmes un personīgās attīstības jomā ir vērojams pieaugošs pieprasījums. Cilvēki arvien vairāk apzinās nepieciešamību nepārtraukti mācīties un pilnveidoties, lai gūtu panākumus un sasniegtu savus mērķus. Šī mājaslapa piedāvās aktuālu un noderīgu informāciju dažādās pašizaugsmes jomās, kas palīdzēs lietotājiem sasniegt savus mērķus.

1. UZDEVUMA NOSTĀDNE

Noslēguma darba uzdevums ir izveidot digitālo grāmatu bibliotēkas mājaslapu, kurā būtu apkopotas grāmatas angļu valodā, fokusējoties uz sevis pilnveidošanu biznesā, veselībā, produktivitātē un vēl. Padarot šīs grāmatas viegli pieejamas un izmantojamas.

Digitālās grāmatas bibliotēku sistēmai būs šādas funkcionalitātes:

- Reģistrācijas funkcionalitāte.
- Lasīt e-grāmatas tajā pašā mājaslapā, saglabāt favorītu sarakstā un lejupielādēt tās.
- Rakstīt atsauksmes par e-grāmatām un komunicēt ar citiem lietotājiem forumos.
- Piezīmju klade, kur var pierakstīt sev svarīgas lietas, ko no grāmatas esi ieguvis.
- Iespēja meklēt pēc nosaukuma un filtrēt pēc žanra vai reitinga e-grāmatas.

1.1 Sistēmas lietotāji un to mijiedarbība ar sistēmu

Lai varētu sākt projektēt šo sistēmu, ir jāsaprot, kā šī sistēma strādās. Lai to saprastu, tika izveidota lietošanas gadījumu diagramma. Šī diagramma parāda un palīdz izprast sistēmas funkcionalitāti un lietotāju vajadzības.



1.1. att. Lietošanas gadījumu diagramma

Viesiem būs piekļuve pie sākuma mājaslapas, kur tiem būs iespēja apskatīt, ko piedāvā šī mājaslapa un reģistrēties tajā. Kamēr viņi nav reģistrējušies, pie grāmatām un pārējām mājaslapas funkcijām viesi nevar piekļūt.

Reģistrētiem lietotājiem būs iespēja piekļūt grāmatu klāstam. Viņi varēs lasīt šīs grāmatas mājaslapā un pierakstīt virtuālajā pierakstu kladē vai lejupielādēt tās. Reģistrēts lietotājs varēs arī komunicēt ar citiem lietotājiem forumos, kur lietotāji varēs diskutēt par kādām grāmatām vai ieteikt cits citam kādu grāmatu. Lietotājs vēl varēs rakstīt atsauksmes par grāmatām un pievienot tās favorītu sarakstam.

Administratoriem būs tādas pašas iespējas, kā reģistrētiem lietotājiem. Papildus administratori varēs piekļūt vadības panelim, kurā būs iespēja augšupielādēt un dzēst grāmatas. Būs arī iespēja pārskatīt reģistrētos lietotājus, piešķirt administratora lomu, rediģēt un dzēst lietotāja datus. Kā arī administratoriem būs iespēja dzēst cita lietotāja publicētās ziņas forumos.

1.2 Uzdevumu risināšanas līdzekļu izvēles pamatojums

Lai izveidotu šo sistēmu, tiks izmantotas dažādas tehnoloģijas, kas padarīs vieglāku sistēmas izstrādi un drošību. Par cik šī bibliotēka būs kā mājaslapa, tās priekšpusei tiks izmantots HTML, CSS un JavaScript. HTML struktūrēs informāciju, CSS piešķirs tai izskatu un stilu, bet JavaScript nodrošinās dinamiku un interaktivitāti. Tas palīdz veidot lietotājiem draudzīgu interfeisu, kur grāmatas var tikt skatītas un pārlūkotas efektīvi.

Lai varētu attēlot grāmatas mājaslapā, tiks izmantots PDF.js. PDF.js nodrošinās ērtu lietošanas pieredzi lietotājiem, kuri vēlas lasīt grāmatas tiešsaistē bez nepieciešamības lejupielādēt un atsevišķi atvērt PDF failus.

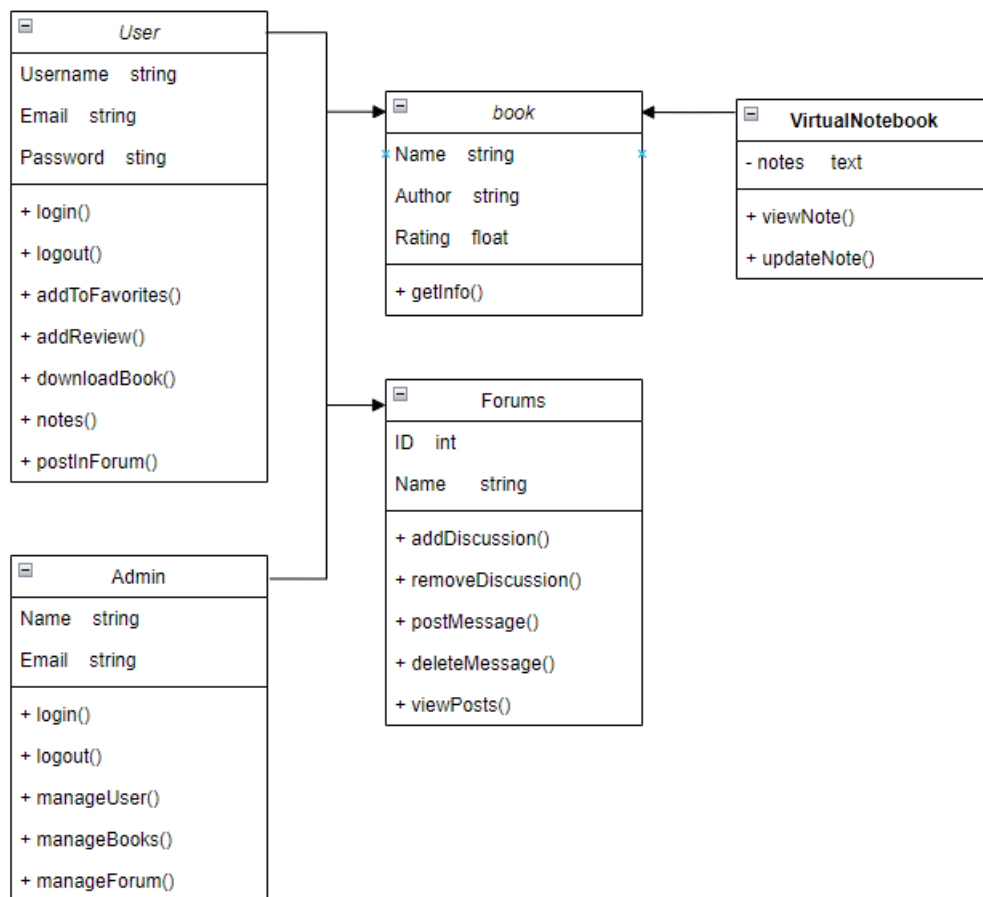
Mājaslapas aizmugurei tiks lietots PHP Laravel. Tas nodrošinās efektīvu un elegantu back-end risinājumu šai mājaslapai, piedāvājot struktūru, drošību un daudzus iebūvētus rīkus, lai veiksmīgi pārvaldītu datu apstrādi un lietotāju autentifikāciju.

Grāmatas tiks glabātas vietējā serverī mapē public/assets. Šī pieeja nodrošina vienkāršāku ieviešanu un ātrāku izstrādi.

Lai varētu šai sistēmai glabāt informāciju, tiks izmantota Laragon MySQL datubāze. Laragon MySQL datubāze nodrošinās drošu un uzticamu datu glabāšanas risinājumu, kas ir viegli lietojams un administrējams, pateicoties iebūvētajai grafiskajai saskarnei.

2. DATU MODELĒŠANA

2.1. Objektu orientēts konceptuālais datu modelis.



2.1 att. Klasu diagramma

Katrs “VirtualNotebook” var saturēt pierakstus, un katrs pieraksts ir saistīts ar konkrētu “Book” objektu. Tas nozīmē, ka pieraksts atbilst konkrētai grāmatai.

Lietotāji var veikt darbības ar dažādām grāmatām, piemēram, apskatīt informāciju, pievienot tās favorītiem vai lejupielādēt. Šeit nav tiešas asociācijas, bet lietotāji veic darbības, izmantojot informāciju no “Book” klases.

Administratori mantos visus atribūtus un metodes no lietotājiem, bet papildus tiem var veikt arī administratīvas darbības, piemēram, pārvaldīt lietotājus vai dzēst ziņas forumā.

Administratori var veikt specifiskas darbības forumā, piemēram, dzēst citas personas ziņojumus vai izveidot galvenās foruma tēmas.

2.2 Entitīju relāciju datu modelis

Projektējot datubāzi tika izstrādāts ER (entitīju relāciju) diagramma, kuru var apskatīt pielikumā (1.pielikums). Diagramma attēlo entitijas, atribūtus un relācijas. Šī diagramma palīdzēs izprast šī projekta datu bāzes dizainu.

- “Users” – Satur lietotāja informāciju. Lietotāja atribūti: id, vārds, e-pasts, parole, lietotāja tips, reģistrēšanās datums.
- “Books” – Satur informāciju par grāmatu. Grāmatu atribūti: id, nosaukums, autors, kategorija, vāka attēls, faila atrašanās vieta, pievienošanas datums.
- “Reviews” – Satur lietotāja atsauksmes informāciju par grāmatām. Atsauksmes atribūti: id, vērtējums, atsauksmes teksts, izveidošanas datums.
- “Favorites” – Satur informāciju par lietotāja mīļākajām grāmatām. Favorītu atribūti: id, saglabāšanas datums.
- “Notes” – Satur informāciju par lietotāja pierakstiem virtuālajā piezīmju kladē. Piezīmju klades atribūti: id, teksts, izveidošanas datums.
- “Forum” – Satur informāciju par foruma tēmām, kur lietotāji var diskutēt. Forumā atribūti: id, nosaukums (tēma), izveidošanas datums.
- “Forum_message” – satur informāciju par lietotāja ziņām forumos. Forumā ziņu atribūti: id, ziņa, publicēšanas datums.

Entitīju relācijas diagrammā ir attēlotas relācijas starp entitijām. Lūk šādi tiek savienotas entitijas:

- Starp tabulām “Users” un “Reviews” ir attiecība viens pret daudziem, jo viens lietotājs var rakstīt vairākas atsauksmes, bet viena atsauksme pieder vienam lietotājam.
- Starp tabulām “Users” un “Notes” ir attiecība viens pret daudziem, jo viens lietotājs var rakstīt vairākas piezīmes sev, bet katra piezīmi izveido viens lietotājs.
- Starp tabulām “Users” un “Favorites” ir attiecība viens pret daudziem, jo vienam lietotājam var būt vairāki favorīti, bet katrs favorīts ir piesaistīts pie viena lietotāja.
- Starp tabulām “Books” un “Reviews” ir attiecība viens pret daudziem, jo vienai grāmatai var būt vairākas atsauksmes, bet katra atsauksme pieder pie vienas grāmatas.
- Starp tabulām “Books” un “Notes” ir attiecība viens pret daudziem, jo vienai grāmatai var būt vairāki pieraksti, bet katrs pieraksts ir saistīts ar konkrētu grāmatu.

- Starp tabulām “Users” un “Forum_messages” ir attiecība viens pret daudziem, jo viens lietotājs var rakstīt vairākas foruma ziņas, bet katra foruma ziņa pieder pie viena lietotāja.
- Starp tabulām “Forums” un “Forum_messages” ir attiecība viens pret daudziem, jo zem viena foruma var būt vairākas ziņas, bet katra ziņa ir piesaistīta pie noteikta foruma.

3. DATU BĀZES IZSTRĀDE

3.1 Datu bāzes fiziskā struktūra

Datu bāzes fiziskā struktūra tika realizēta izmantojot ER diagrammu. Projektēšanas laikā tika izveidotas 7 tabulas, kas satur informāciju par grāmatām, lietotājiem, lietotāju pierakstiem, lietotāju grāmatu favorītiem, atsauksmēm, forumiem un forumu ziņām. Datu bāzes tabulu shēmu var apskatīt pielikumā (2. Pielikums).

Tabula “Users” satur reģistrētā lietotāja datus: id (unikāls katram lietotājam), vārds, e-pasts, parole, lietotāja tips, reģistrēšanās datums.

3.1. tabula

Tabulas “Users” struktūra

Nr.	Lauka nosaukums	Datu tips	Izmērs	Apraksts
1.	ID	Int	-	Primārā atslēga
2.	Name	Varchar	255	Lietotāja vārds
3.	Email	Varchar	255	Lietotāja e-pasts
4.	Password	Varchar	255	Lietotāja parole
5.	Uertype	Varchar	255	Lietotāja tips
6.	Created_at	Timestamp	-	Reģistrācijas laiks
7.	Updated_at	Timestamp	-	Atjaunošanas laiks

Tabula “Books” satur informāciju par grāmatām: id (unikāls katrai grāmatai), nosaukums, autors, kategorija, vāka bilde, augšupielādes datums.

3.2. tabula

Tabulas “Books” struktūra

Nr.	Lauka nosaukums	Datu tips	Izmērs	Apraksts
1.	ID	Int	-	Primārā atslēga
2.	Title	Varchar	255	Grāmatas nosaukums
3.	Author	Varchar	255	Grāmatas autors
4.	Category	Varchar	255	Grāmatas kategorija
5.	Cover_Image	BLOB	-	Grāmatas vāka bilde
6.	Uploaded_at	Timestamp	-	Augšupielādes datums

Tabula “Reviews” satur grāmatas atsauksmes, ko lietotāji ir rakstījuši: id (unikāls katrai atsauksmei), grāmatas vērtējums, atsauksmes teksts, publicēšanas datums, lietotāja ID un grāmatas ID (ārējās atslēgas).

Tabulas “Reviews” struktūra

Nr.	Lauka nosaukums	Datu tips	Izmērs	Apraksts
1.	ID	INT	-	Primārā atslēga
2.	Review_score	INT	255	Lietotāja vārds
3.	Review_text	Varchar	255	Lietotāja e-pasts
4.	Created_at	TIMESTAMP	-	Publicēšanas laiks
5.	Users_ID	INT	-	Lietotāja ID (ārējā atslēga)
6.	Books_ID	INT	-	Grāmatas ID (ārējā atslēga)

Tabula “Favorites” satur lietotāja mīļākas grāmatas: id (unikāls katram favorītam), izveidošanas datums, lietotāja un grāmatas identifikatori (ārējās atslēgas).

Tabulas “Favorites” struktūra

Nr.	Lauka nosaukums	Datu tips	Izmērs	Apraksts
1.	ID	INT	-	Primārā atslēga
2.	Created_at	TIMESTAMP	-	Pievienošanas laiks
3.	Users_ID	INT	-	Lietotāja ID (ārējā atslēga)
4.	Books_ID	INT	-	Grāmatas ID (ārējā atslēga)

Tabula “Notes” satur lietotāja piezīmes (pierakstus) pie katras grāmatas: id (unikāls katram pierakstam), piezīmes teksta autors, izveidošanas datums, lietotāja un grāmatas identifikatori (ārējās atslēgas).

Tabulas “Notes” struktūra

Nr.	Lauka nosaukums	Datu tips	Izmērs	Apraksts
1.	ID	INT	-	Primārā atslēga
2.	Note_text	TEXT		Piezīmju teksts
3.	Created_at	TIMESTAMP	-	Pievienošanas laiks
4.	Users_ID	INT	-	Lietotāja ID (ārējā atslēga)
5.	Books_ID	INT	-	Grāmatas ID (ārējā atslēga)

Tabula “Forums” satur foruma tēmas, kur lietotāji varēs diskutēt: id (unikāls katram forumam), foruma nosaukums, izveidošanas datums, lietotāja identifikators (ārējā atslēga).

Tabulas “Forums” struktūra

Nr.	Lauka nosaukums	Datu tips	Izmērs	Apraksts
1.	ID	INT	-	Primārā atslēga
2.	Title	Varchar	255	Foruma tēma
3.	Created_at	TIMESTAMP	-	Pievienošanas laiks
4.	Users_ID	INT	-	Lietotāja ID (ārējā atslēga)

Tabula “Forum_messages” satur lietotāja ziņas forumos: id (unikāls katrai ziņai), lietotāja ziņa, publicēšanas datums, lietotāja un foruma identifikātors (ārējās atslēgas).

Tabulas “Forum_messages” struktūra

Nr.	Lauka nosaukums	Datu tips	Izmērs	Apraksts
1.	ID	INT	-	Primārā atslēga
2.	Message	TEXT	-	Ziņa
3.	Created_at	TIMESTAMP	-	Publicēšanas datums
4.	Users_ID	INT	-	Lietotāja ID (ārējā atslēga)
5.	Forums_ID	INT	-	Foruma ID (ārējā atslēga)

3.2 Datu bāzes optimizēšana

Lai šī sistēma būtu labi strādājoša un veiksmīga, ir būtiski, lai datu bāze būtu labi optimizēta. Optimizēšana ietver vairākus aspektus, tai skaitā piemērotu datu tipu izvēli katram laukam, lai samazinātu datu apjomu un uzlabotu datu pārstrādes veikspēju.

Katram laukam datu bāzē ir jāizvēlas piemērots datu tips, kas atbilst datu raksturam un atļauj samazināt datu apjomu. Piemēram, ID laukiem, kas izmantojas kā primārās atslēgas vai ārējās atslēgas, var izmantot veselu skaitli (integer), lai samazinātu atmiņas izmantošanu, bet teksta laukiem, piemēram, vārdiem un epastiem, var izmantot varchar ar atbilstošu garumu.

Katrai tabulai ir jābūt ar vienu primāro atslēgu, kas viennozīmīgi identificē ierakstus šajā tabulā. Šis princips palīdz nodrošināt efektīvu datu piekļuvi.

Katrs tabulas lauks ir funkcionāli atkarīgs no primārās atslēgas, tādējādi nodrošinot visu informāciju par katru ierakstu noteiktā tabulā.

4. SISTĒMAS FUNKCIONĀLO PRASĪBU REALIZĒŠANA

4.1. Datu pievienošana, labošana, dzēšana

Sistēmā ir iespēja pievienot jaunu lietotāju un grāmatu. Labot lietotāja un grāmatas datus, kā arī dzēst tos. Lietotājs tiek pievienots izmantojot reģistrācijas lapu, kurā viņš norāda savu lietotājvārdu, e-pastu un paroli. Pēc to aizpildīšanas, sistēma veic pārbaudi pēc noteiktiem kritērijiem un pievieno lietotāju “users” tabulai. Šajā tabulā ir ID, lietotājvārds, e-pasts, šifrēta parole, lietotāju tips, izveidošanas laiks (kad pieteikās) un pēdējoreiz atjaunināts profils. 4.1 attēlā ir redzams tabulas “users” saturs pēc lietotāja reģistrācijas.

id	name	email	password	usertype	created_at	updated_at
1	admin	admin@gmail.	\$2y\$12\$B7cL7FKKW0vAM	admin	2024-03-27 18:10:47	2024-03-27 18:10:47
2	User	user@gmail.c	\$2y\$12\$8hbdBZqsykGyE	user	2024-03-27 18:13:28	2024-03-27 18:13:28
3	tests	tests@gmail.c	\$2y\$12\$/yZ.b0ly30paxns1	user	2024-04-10 14:11:32	2024-04-10 16:02:27
4	baton	phone@gmail.	\$2y\$12\$ebysNaFwy70bC4	user	2024-04-13 15:03:09	2024-04-13 15:03:09

4.1. att. Tabulas “users” saturs

Lietotājs var mainīt savu e-pastu un paroli vai arī dzēst savu profilu. Skatīt visas pieejamās grāmatas, rakstīt atsauksmes par tām. Lasot grāmatas lietotājs var pierakstīt sev svarīgas piezīmes jebkurā laikā. Atsevišķa “notes” lapā, lietotājs var pārskatīt visas piezīmes, ko ir veicis, kā arī dzēst tās un doties uz grāmatu, kur piezīme tika veikta. Lietotājam ir iespēja saglabāt un noņemt grāmatas no favorītu saraksta.

Administratoriem ir iespēja piešķirt lietotājam administratora tiesības un dzēst pašus lietotājus. Lai demonstrētu funkcionalitāti, no administratora paneļa, piešķirs lietotājam ar id 3 administratora lietotāja tipu un izdzēsīs lietotāju ar id 4 skatīt 4.2. attēlu.

id	name	email	password	usertype	created_at	updated_at
1	admin	admin@gmail.com	\$2y\$12\$B7cL7FKKW0vAWPgK2oqbbebZzsqcL3UdjO1OesCKi59...	admin	2024-03-27 18:10:47	2024-03-27 18:10:47
2	User	user@gmail.com	\$2y\$12\$8hbdBZqsykGyEmGA4GysOGxcRdJh1cdxqpGc0ChZkS...	user	2024-03-27 18:13:28	2024-03-27 18:13:28
3	tests	tests@gmail.com	\$2y\$12\$/yZ.b0ly30paxns1fhS01OiLCCtnZwT0Hb4L/Q1bxGu...	admin	2024-04-10 14:11:32	2024-04-30 22:08:33

4.2. att. Tabulas “users” saturs

4.2. Datu validācija

Lietotājam veidojot profilu notiek datu validācija. Katram ievades laukam ir pievienots parametrs “required”, kas liek lietotājam obligāti aizpildīt katru lauku. Šis required parametrs tiek izmantots pieslēgšanās lapai un grāmatu augšupielādes lapā. Vēl reģistrācijas lapā tiek pārbaudīts vai e-pastā ir @ simbols un vai parole sastāv ne mazāk par 8 simboliem. Augšupielādējot grāmatas, tiek pārbaudīts vai fails ir .pdf formātā un nepārsniedz 10 mb izmēru.

4.3. Datu izgūšana un sapstrāde

Reģistrētie lietotāji var apskatīt visas pieejamās grāmatas, savu favorītu sarakstu un pierakstus. Pierakstu lapas sadaļā tiek norādīts grāmatas nosaukums, zem kura šis pieraksts ir veikts, un datums, kas norāda pēdējās pieraksta izmaiņas. Turpat ir arī poga uz kuras uzspiežot, lietotāju novirzīs uz grāmatas lapu, kur šis pieraksts tika veikts. Administratori var darīt to pašu, ko lietotāji un vēl pārvaldīt lietotājus, augšupielādēt, rediģēt un dzēst grāmatas. 4.3, 4.4 un 4.5 attēlā ir redzams, kā šie dati tiek attēloti mājaslapā. Grāmatu bildes tiek iegūtas izmantojot pdf.js bibliotēku, kas paņem grāmatas pirmo lapu, kā bildi.

ADMIN

USER MANAGEMENT

SEARCH FOR NAMES...

FILTER NAME A-Z

ID	NAME	EMAIL	USER TYPE	CREATED AT	UPDATED AT	EDIT	DELETE
1	admin	admin@gmail.com	Admin	2024-03-27 18:10:47	2024-03-27 18:10:47	Edit	Delete
2	User	user@gmail.com	User	2024-03-27 18:13:28	2024-03-27 18:13:28	Edit	Delete
3	tests	tests@gmail.com	Admin	2024-04-10 14:11:32	2024-04-30 22:08:33	Edit	Delete

4.3. att. Lietotāju pārvaldības lapā

NOTES

Book: Velviena gramata
Last updated 17 minutes ago

wrtuighkjoygkjghnrtghjeglkretg

Book: testings
Last updated 18 days ago

4.4. att. Grāmatu pierakstu lapā

Goals!

How to Get Everything You Want – Faster Than You Ever Thought Possible

By: Brian Tracy

The Unfair Advantage

How You Already Have What It Takes to Succeed

ASH ALI & HASAN KUBBA

MAN'S SEARCH FOR HIMSELF

BOLLO MAY

Tiny Changes, Remarkable Results

Atomic Habits

Title: Atomic Habits
Author: James Clear
Category: Productivity

NEVER SPLIT THE DIFFERENCE

NEGOTIATING AS IF YOUR LIFE DEPENDED ON IT

CHRIS VOSS

4.5. att. Grāmatu pārvaldība

14

4.4. Datu integritātes nodrošināšanas metožu izmantošana

4.4.1. Lietotāju autentifikācija

Lai iekļūtu mājaslapā, lietotājiem vispirms ir jāpiesakās. Dažas lapas ir aizsargātas ar autentifikācijas starpprogrammatūru (middleware). Tas nodrošina, ka tikai autentificēti lietotāji ar derīgām sesijām (sessions) var skatīt un mijiedarboties ar vietni. Veiksmīga pieteikšanās sāk šifrētu sesiju serverī, un aizsargātās lapās tiek izmantota autentifikācijas starpprogrammatūra, lai pirms piekļuves atļaujas pārbaudītu, vai sesija ir derīga. Sesijas beidzas pēc neaktivitātes perioda vai, ja lietotājs iziet (logout), un lietotājiem ir atkārtoti jāautentificējas, lai piekļūtu mājaslapas saturam. Reģistrēšanās prmskodu var redzēt 3. Pielikumā un autentifikācijas prmskodu 4. Pielikumā.

4.4.2. Funkcionālo pienākumu sadalīšana

Reģistrētiem lietotājiem var būt divas lomas, parasts lietotājs vai administrators. Kad jauns lietotājs pirmo reizi pierēģistrējas, viņam automātiski tiek iedota parastā lietotāja loma.

```
$table->string('usertype')->default('user');
```

4.6. att. Lietotāja lomu piešķiršana

Šī loma tiek glabāta datubāzes tabulā “users” zem lauka “usertype”. Kad lietotājs ieiešanas lapā ievada savus datus un nospiež pogu ieiet, tiek palaists kods (skatīt 4.7. attēlu), kas pārbauda lietotāja tipu, un pēc tā, lietotājs attiecīgi tiek novirzīts uz savu lapu. Parasts lietotājs tiek novirzīts uz lietotāja paneli un administrators uz administratora paneli.

```
public function index()
{
    if (Auth::id()) {
        $usertype = Auth()->user()->usertype;

        if ($usertype == 'user') {
            return view('dashboard');
        } else if ($usertype == 'admin') {
            return view('admin.adminhome');
        } else {
            return redirect()->back();
        }
    }
}
```

4.7. att. Lietotāja novirzīšana pēc lomas

NOBEIGUMS

Nobeigumā var secināt, ka projekta mērķis ir gandrīz izdevies. Ir sistēma, kas ļauj lietotājiem pārlūkot un lasīt digitālās grāmatas. Lietotāji var rakstīt piezīmes un atsauksmes par šīm grāmatām. Ir meklēšanas un filtrēšanas funkcijas, kas ļauj lietotājiem ērtāk atrast grāmatas.

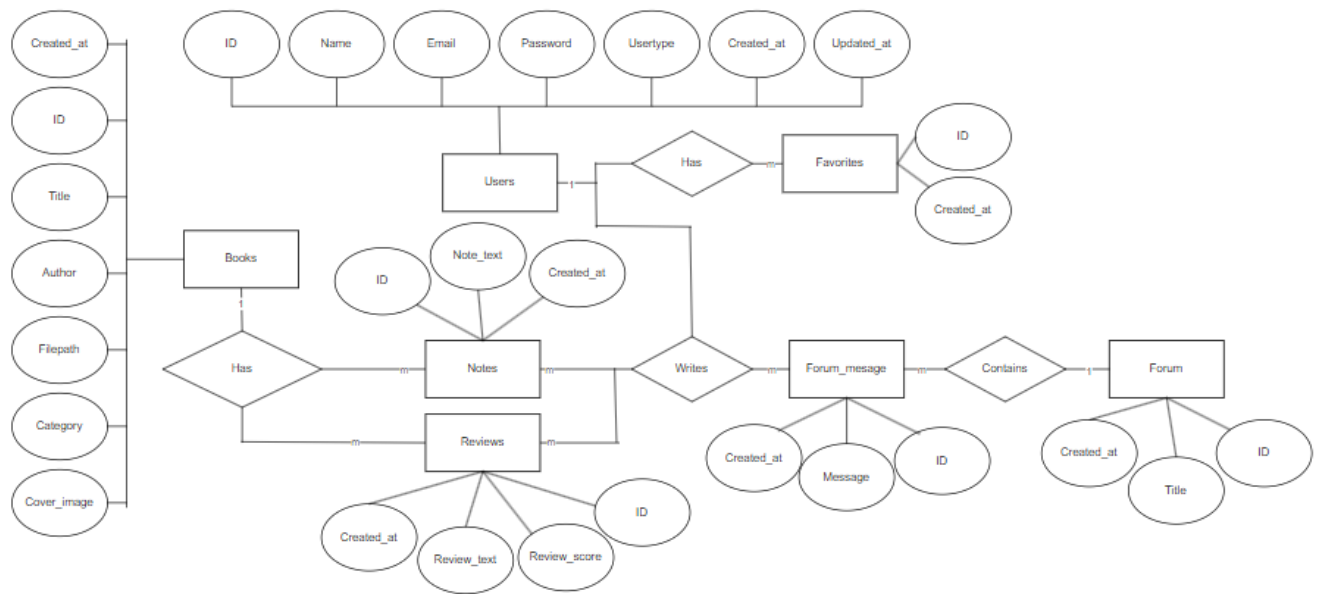
Nākamajā attīstības posmā plānots ieviest sociālo funkciju, kas ļaus lietotājiem veidot diskusiju forumus, dalīties ar grāmatu ieteikumiem un veidot kopienu. Papildus tam, tiks ieviestas personalizācijas iespējas, piemēram, grāmatu ieteikumu sistēma un individuāli lasīšanas mērķi.

Šī projekta izstrāde ir bijusi vērtīga pieredze, kas ļāva man apgūt jaunas tehnoloģijas, kā, Laravel PHP frameworku un PDF.js bibliotēku, kā arī pilnveidot datu bāzu zināšanas.

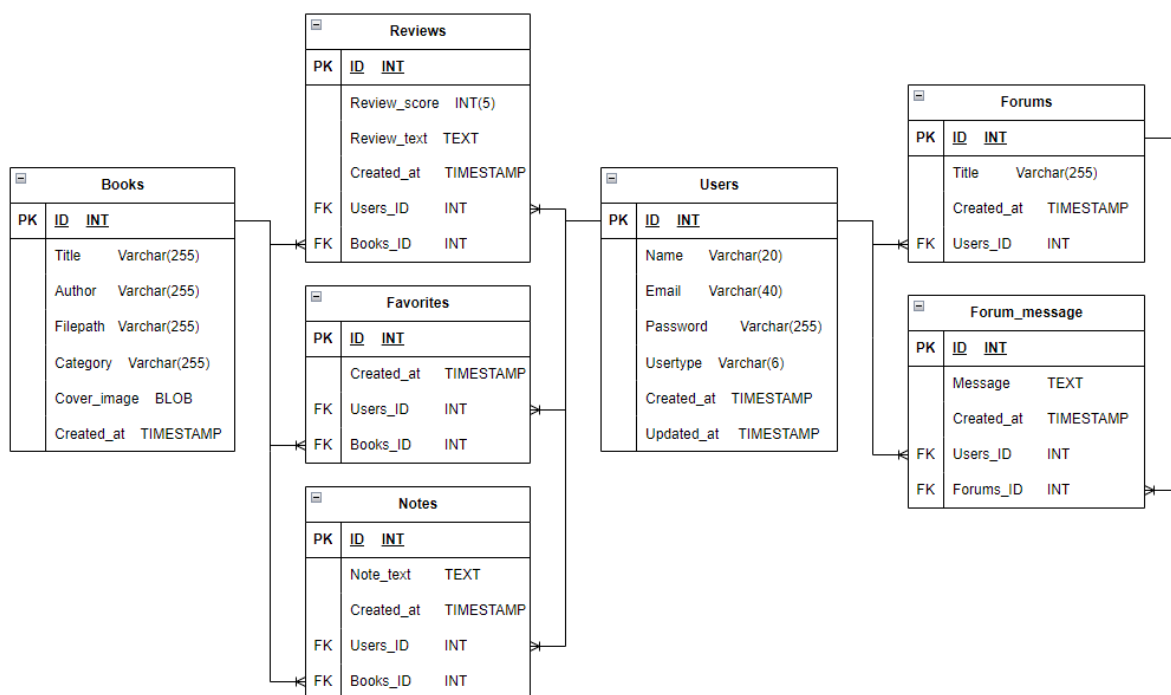
INFORMĀCIJAS AVOTI

1. Laravel dokumentācija - <https://laravel.com/docs/10.x> (Resurss apskatīts 04.05.2024.)
2. Pdf.js bibliotēka - <https://github.com/mozilla/pdf.js> (Resurss apskatīts 04.05.2024.)
3. CSS - https://www.w3schools.com/css/css_grid.asp (Resurss apskatīts 04.05.2024.)
4. HTML - <https://www.w3schools.com/html/default.asp> (Resurss apskatīts 04.05.2024.)
5. Laragon dokumentācija - <https://laragon.org/docs/index.html> (Resurss apskatīts 04.05.2024.)
6. JS - https://www.w3schools.com/js/js_ajax_intro.asp (Resurss apskatīts 04.05.2024.)
7. StackOverflow - <https://stackoverflow.com/questions/49248709/laravel-undefined-variable-data-on-view> (Resurss apskatīts 04.05.2024.)
8. Kļāšu Diagramma - <https://www.lucidchart.com/pages/uml-class-diagram> (Resurss apskatīts 04.05.2024.)

Datu bāzes ER modelis



Datu bāzes tabulu shēma



Reģistrācijas pirmkods

```

3 references | 0 implementations
class RegisteredUserController extends Controller
{
    1 reference | 0 overrides
    public function create(): View
    {
        return view('auth.register');
    }

    /**
     * Handle an incoming registration request.
     *
     * @throws \Illuminate\Validation\ValidationException
     */
    1 reference | 0 overrides
    public function store(Request $request): RedirectResponse
    {
        $request->validate([
            'name' => ['required', 'string', 'max:255'],
            'email' => ['required', 'string', 'lowercase', 'email', 'max:255', 'unique:'.User::class],
            'password' => ['required', 'confirmed', Rules\Password::defaults()],
        ]);

        $user = User::create([
            'name' => $request->name,
            'email' => $request->email,
            'password' => Hash::make($request->password),
        ]);

        event(new Registered($user));

        Auth::login($user);

        return redirect(RouteServiceProvider::HOME);
    }
}

```

Autentifikācijas pirmskods

```
4 references | 0 implementations
class AuthenticatedSessionController extends Controller
{
    /**
     * Display the login view.
     */
    1 reference | 0 overrides
    public function create(): View
    {
        return view('auth.login');
    }

    /**
     * Handle an incoming authentication request.
     */
    1 reference | 0 overrides
    public function store(LoginRequest $request): RedirectResponse
    {
        $request->authenticate();

        $request->session()->regenerate();

        return redirect()->intended(RouteServiceProvider::HOME);
    }

    /**
     * Destroy an authenticated session.
     */
    1 reference | 0 overrides
    public function destroy(Request $request): RedirectResponse
    {
        Auth::guard('web')->logout();

        $request->session()->invalidate();

        $request->session()->regenerateToken();

        return redirect('/');
    }
}
```