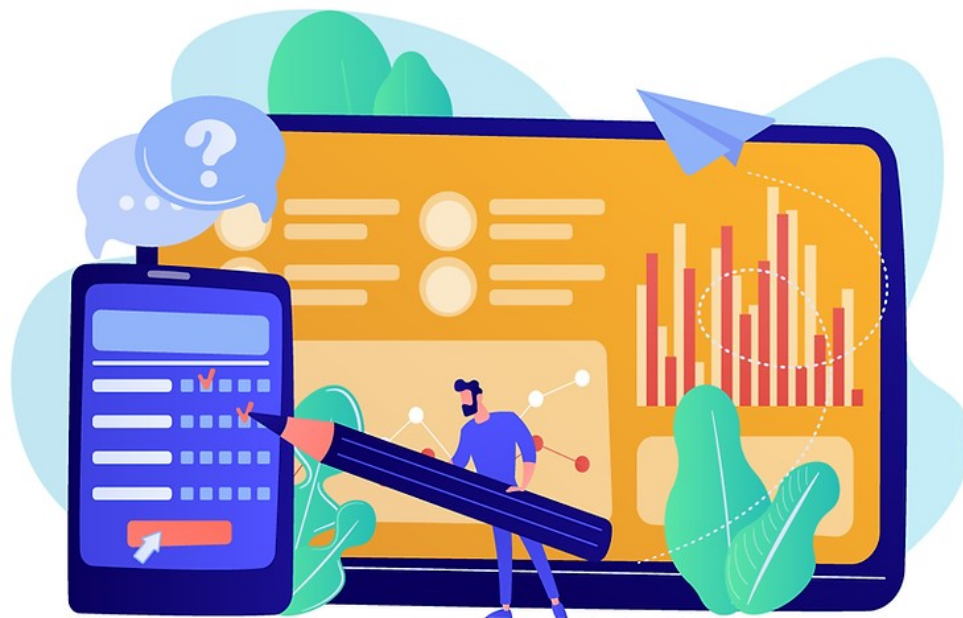


Projecte 5 - Cendrasoft, Solucions Web



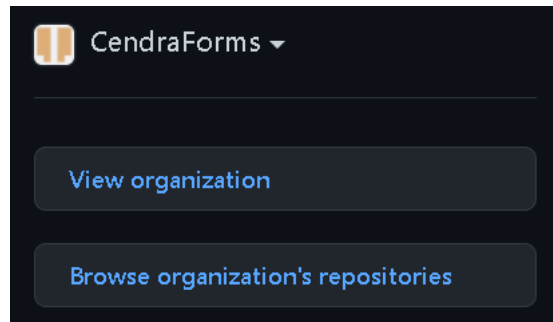
- Pàgina d'Informació: <https://docs.google.com/>
- Projecte GitHub: <https://www.github.com/>
- Membres Del Grup:
 - Àlex Ramon
 - Ben Younes Bouhtat Diba
 - Bernat Ruiz Cros
 - Eimantas Milkintas
 - Eric Moreno Pou
 - Juan José Gomez Villegas
 - Zaytouny Hicham Araid Ez

Índex

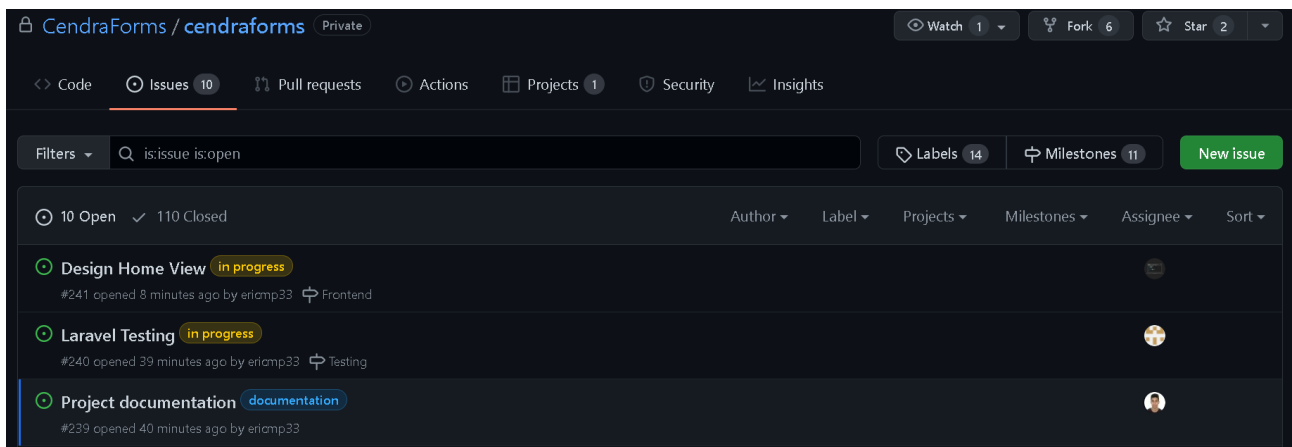
Planificació General.....	3
Anàlisi Dels Requisits.....	5
Manual d'Instal·lació.....	6
Diagrama De Casos D'ús.....	13

Planificació General

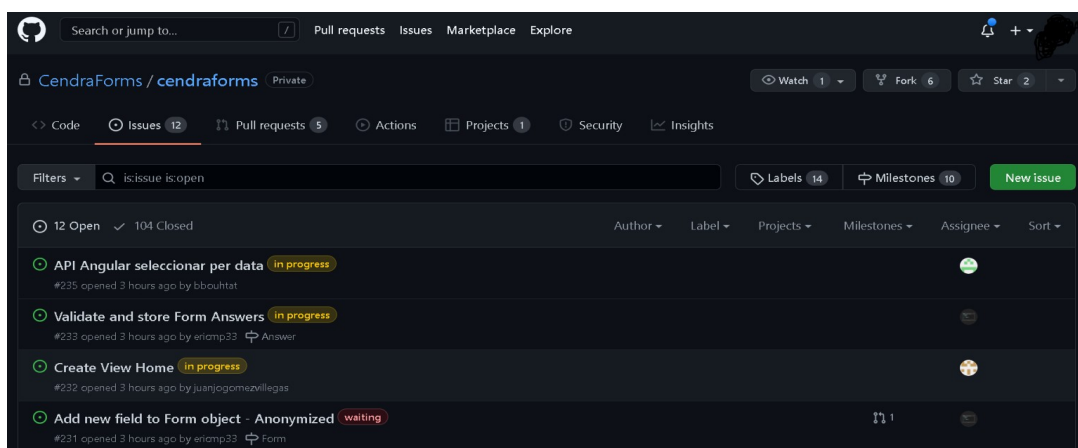
1. Creem una organització del grup on estarà el repositori central:



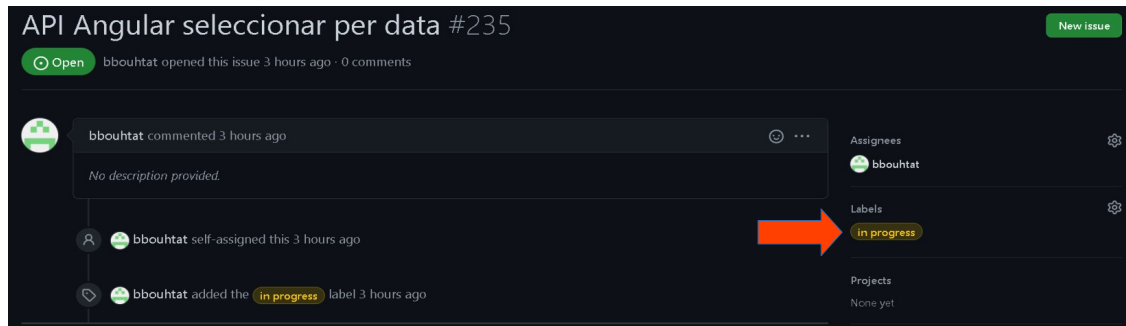
2. Entre el grup ens anem organitzant fent issues de manera conjunta.



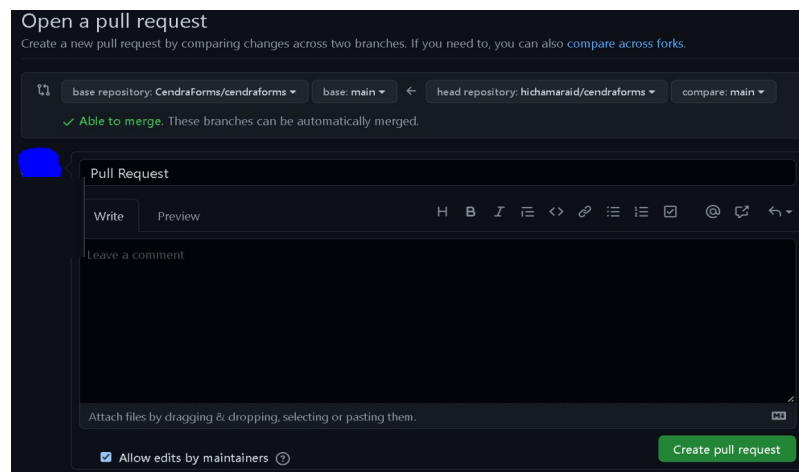
3. Primer el scrum crea varies issues de la mateixa categoria, perquè tothom pugui assignar-se una d'aquell tema en concret.



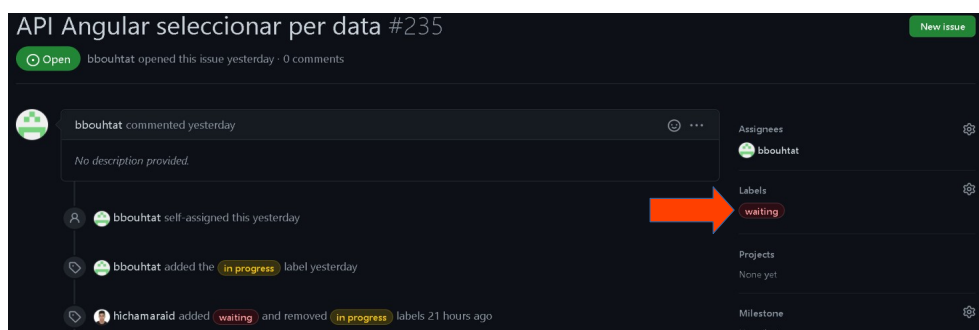
4. Quant un del grup s'assigna una issue té que marcar-la com a in progress perquè el scrum màster sapigui que s'està treballant en aquella issue en concret.



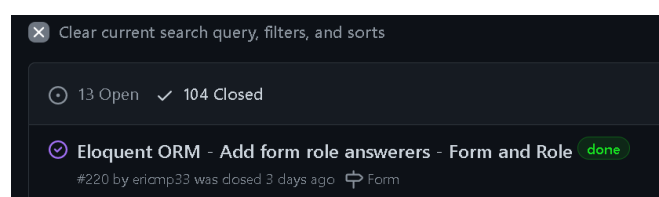
5. Quant el company del grup consideri que ha acabat la issue en concret. Lo que farà es crear una pull-request amb la branca corresponent perquè el scrum màster pugui verificar els errors.



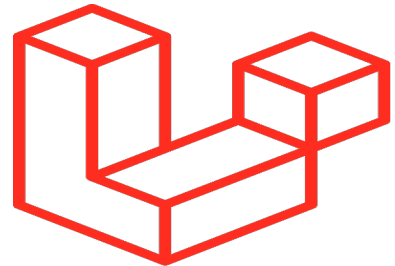
6. Ara només té que tornar a la issue en concret i canviar l'estat de: **in progress** a **waiting**.



7. Ara només té que esperar a que el scrum màster, verifiqui si la issue feta està bé o no, com a conseqüència acceptarà la pull-request o la rebutjarà.

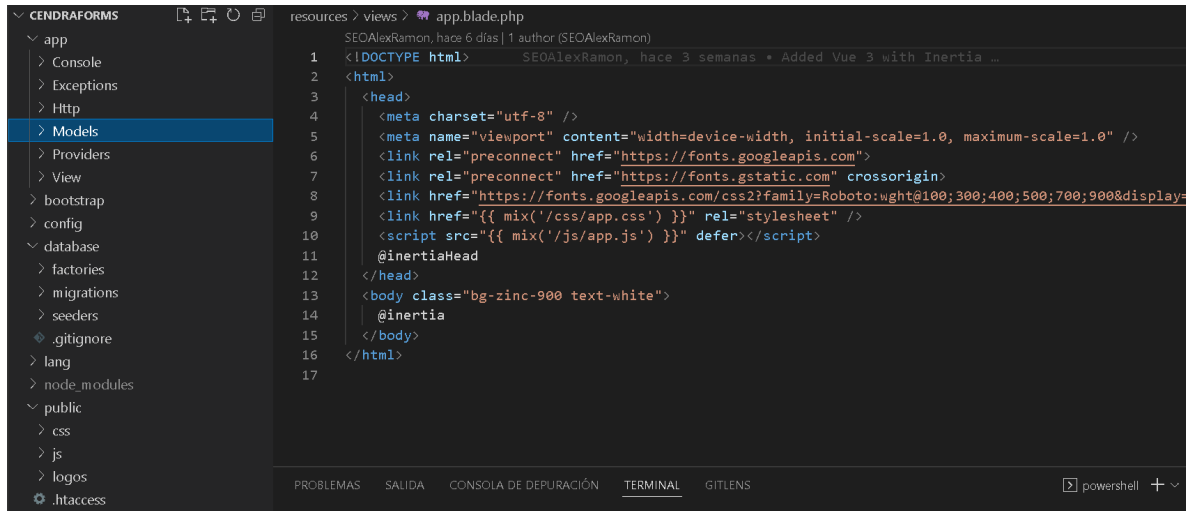


Anàlisi Dels Requisits



Manual d'Instal·lació

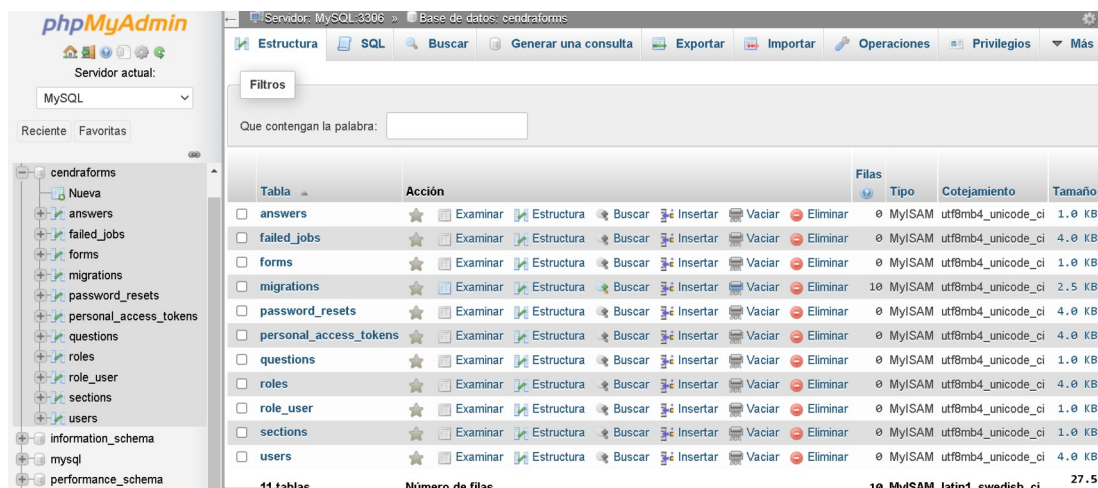
1. Creem un projecte laravel: `composer create-project laravel/laravel cendraforms`



2. Ara Instal·lem els requisits necessaris mencionats anteriorment: **vue, inertia, tailwind, php, etc...**

```
Generating optimized autoload files
> Illuminate\Foundation\ComposerScripts::postAutoloadDump
> @php artisan package:discover --ansi
Discovered Package: inertiajs/inertia-laravel
Discovered Package: laravel/breeze
Discovered Package: laravel/sail
Discovered Package: laravel/sanctum
Discovered Package: laravel/socialite
Discovered Package: laravel/tinker
Discovered Package: nesbot/carbon
Discovered Package: nunomaduro/collision
Discovered Package: socialiteproviders/manager
Discovered Package: spatie/laravel-ignition
Package manifest generated successfully.
79 packages you are using are looking for funding.
Use the `composer fund` command to find out more!
> @php artisan vendor:publish --tag=laravel-assets --ansi --force
No publishable resources for tag [laravel-assets].
Publishing complete.
```

3. Ara ens falta una base de dades per guardar la base de dades, creem una base de dades que es digui cendraforms amb el gestor de base de dades **PhpMyAdmin**:



4. Després de tot aquest procés, ara ens anem al nostre IDE VSCODE en el nostre cas, i obrim una finestra powershell, en la que executarem les següents comandes:

- composer install
- composer update

(Aquestes comandes instal·len i actualitzen els requisits perquè funcioni l'aplicació)

```
Installing dependencies from lock file (including require-dev)
Verifying lock file contents can be installed on current platform.
Nothing to install, update or remove
Generating optimized autoload files
> Illuminate\Foundation\ComposerScripts::postAutoloadDump
> @php artisan package:discover --ansi
Discovered Package: inertiajs/inertia-laravel
Discovered Package: laravel/breeze
Discovered Package: laravel/sail
Discovered Package: laravel/sanctum
Discovered Package: laravel/socialite
Discovered Package: laravel/tinker
Discovered Package: nesbot/carbon
Discovered Package: nunomaduro/collision
Discovered Package: socialiteproviders/manager
Discovered Package: spatie/laravel-ignition
Package manifest generated successfully.
79 packages you are using are looking for funding.
Use the `composer fund` command to find out more!
```

5. Executem la comanda: **php artisan migrate**, per carregar la base de dades amb les taules creades.

Desde la secció de migrations que tenim per defecte i creades.

Per actualitzar la taula de la base de dades utilitzem la comanda: **php artisan migrate:fresh**

```
Dropped all tables successfully.
Migration table created successfully.
Migrating: 2014_10_12_000000_create_users_table
Migrated: 2014_10_12_000000_create_users_table (21.56ms)
Migrating: 2014_10_12_100000_create_password_resets_table
Migrated: 2014_10_12_100000_create_password_resets_table (27.34ms)
Migrating: 2019_08_19_000000_create_failed_jobs_table
Migrated: 2019_08_19_000000_create_failed_jobs_table (21.89ms)
Migrating: 2019_12_14_000001_create_personal_access_tokens_table
Migrated: 2019_12_14_000001_create_personal_access_tokens_table (41.73ms)
Migrating: 2022_02_23_154250_create_roles_table
Migrated: 2022_02_23_154250_create_roles_table (29.20ms)
Migrating: 2022_02_23_155756_create_questions_table
Migrated: 2022_02_23_155756_create_questions_table (25.64ms)
Migrating: 2022_02_23_174456_create_answers_table
Migrated: 2022_02_23_174456_create_answers_table (38.97ms)
Migrating: 2022_02_24_143605_create_sections_table
Migrated: 2022_02_24_143605_create_sections_table (44.03ms)
Migrating: 2022_02_24_155134_create_forms_table
Migrated: 2022_02_24_155134_create_forms_table (22.98ms)
Migrating: 2022_02_25_142918_create_role_user_table
Migrated: 2022_02_25_142918_create_role_user_table (40.99ms)
Migrating: 2022_03_25_180936_create_form_role_answerer_table
Migrated: 2022_03_25_180936_create_form_role_answerer_table (46.78ms)
Migrating: 2022_03_25_181021_create_form_role_editor_table
Migrated: 2022_03_25_181021_create_form_role_editor_table (32.52ms)
```

6. Ara tenim el nostre projecte a punt per començar a treballar. Comencen primer en crear els controllers que creiem necessaris.

```

83     ->with('message', 'answer successfully sent');
84 }
85
86 /**
87  * Validates parsed request.
88  *
89  * @param Request $request Validated request.
90  * @return \Illuminate\Contracts\Validation\Validator
91  */
92 private function validateRequest(Request $request): \Illuminate\Contracts\Validation\Validator
93 {
94     return Validator::make($request->all(), [
95         'form.id' => ['required', 'integer'],
96         'form.published' => ['required', 'boolean'],
97
98         'form.sections' => ['required', 'array', 'min:1'],
99         'form.sections.*.questions' => ['required', 'array', 'min:1'],
100
101         'form.sections.*.questions.*.id' => ['required', 'integer'],
102         'form.sections.*.questions.*.answer' => ['required', 'string_or_array'],
103     ]);
104 }
105
106 /**
107  * Gets all answers
108  *
109  * @return JSON All obtained answers
110  */
111 public static function getAll()
112 {

```

7. Ara tenim que crear els models necessaris per establir les regles de connexió amb la base de dades del projecte.

```

4
5 use Illuminate\Database\Eloquent\Factories\HasFactory;
6 use Illuminate\Database\Eloquent\Model;
7 use Illuminate\Database\Eloquent\Relations\BelongsTo;
8 use Illuminate\Database\Eloquent\Relations\HasMany;
9
10 class Section extends Model
11 {
12     use HasFactory;
13
14     /**
15      * The attributes that are mass assignable.
16      *
17      * @var array
18      */
19     protected $fillable = [
20         'name',
21         'form_id',
22         'user_id',
23     ];
24
25     /**
26      * The attributes that should be hidden for arrays.
27      *
28      * @var array
29      */
30     protected $hidden = [
31         'created_at',
32         'updated_at'
33     ];

```


8. Un de les coses mes importants del projecte és implementar l'autenticació **oAuth** en diversos serveis online com: **google, discord, github, gitlab ...**



9. Ara tenim que afegir les rutes necessaris del projecte en dos principals fitxers:

- **api.php**
- **web.php**

```
web.php
28
29 Route::get('/formulari/crean', [FormController::class, 'creat
30     ->name('form.create')
31     ->middleware('auth');
32
33 Route::get('/formulari/{form}/editar', [FormController::clas
34     ->name('form.edit')
35     ->middleware('auth');
36
37 Route::post('/formulari/{form}/editar', [FormController::clas
38     ->name('form.store')
39     ->middleware('auth');
40
41 Route::get('/formulari/{form}', [FormController::class, 'ansv
42     ->name('form.answer')
43     ->middleware('auth');
44
45 Route::post('/formulari/{form}', [AnswerController::class, 's
46     ->name('answer.store')
47     ->middleware('auth');
48
49 // todo - temporal

api.php
67
68
69 /**
70  * Questions
71  */
72 // Route::get('/questions', [QuestionController::class, 'getA
73
74 Route::get('/questions/{question}', [QuestionController::clas
75
76 // Route::post('/questions', [QuestionController::class, 'cre
77
78 Route::put('/questions/{question}', [QuestionController::clas
79
80 Route::delete('/questions/{question}', [QuestionController::c
81
82
83 /**
84  * Answers
85  */
86 Route::get('/answers', [AnswerController::class, 'getAll']);
87
88 Route::get('/answers/{answer}', [AnswerController::class, 'ge
89
```

10. També comprovarem el funcionament de l'aplicació amb testos unitaris en la qual passen com una mena de proves establertes per el programador que ha fet els testos unitaris i dona un resultat del test.

```
PASS Tests\Unit\ExampleTest
✓ that true is true

FAIL Tests\Feature\Auth\AuthenticationTest
✓ login screen can be rendered
✗ users can authenticate using the login screen
✗ users can not authenticate with invalid password

FAIL Tests\Feature\Auth\EmailVerificationTest
✗ email verification screen can be rendered
✗ email can be verified
✗ email is not verified with invalid hash
```

```
FAIL Tests\Feature\Auth\PasswordConfirmationTest
✗ confirm password screen can be rendered
✗ password can be confirmed
✗ password is not confirmed with invalid password

FAIL Tests\Feature\Auth\PasswordResetTest
✓ reset password link screen can be rendered
✗ reset password link can be requested
✗ reset password screen can be rendered
✗ password can be reset with valid token

PASS Tests\Feature\Auth\RegistrationTest
✓ registration screen can be rendered
```

11. També creem Seeders per omplir la base de dades amb una informació ja pre-establerta:

```
EXPLORADOR
CENDRAFORMS
  seeders
    AnswerSeeder.php
    DatabaseSeeder.php
    FormRoleAnswererSeeder.php
    FormRoleEditorSeeder.php
    FormSeeder.php
    QuestionSeeder.php
    RoleSeeder.php
    SectionSeeder.php
    UserSeeder.php
  .gitignore
  lang
  node_modules
  public
    css
    js
    logos
  .htaccess
  favicon.ico
  index.php
  logo.png
  main.js
  mix-manifest.json
  robots.txt

AnswerSeeder.php
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46

'user_id' => 1,
]);

Answer::create([
    'content' => 'Et posaria un 5',
    'question_id' => 1,
    'user_id' => 2,
]);

Answer::create([
    'content' => 'Sí, has resolt els meus dubtes',
    'question_id' => 2,
    'user_id' => 1,
]);

Answer::create([
    'content' => 'No, no has resolt els meus dubtes',
    'question_id' => 2,
    'user_id' => 2,
]);

Answer::create([
    'content' => 'Sí, ets un bon professor',
    'question_id' => 3,
    'user_id' => 2,
]);
```

- Exemple d'execució d'un Seeder:

```
php artisan db:seed --class=RoleSeeder
Database seeding completed successfully.
```

+ Opciones

						id	name	active	created_at	updated_at	
<input type="checkbox"/>		Editar		Copiar		Borrar	1	direccio	1	2022-04-01 16:32:48	2022-04-01 16:32:48
<input type="checkbox"/>		Editar		Copiar		Borrar	2	professor	1	2022-04-01 16:32:48	2022-04-01 16:32:48
<input type="checkbox"/>		Editar		Copiar		Borrar	3	alumne	1	2022-04-01 16:32:48	2022-04-01 16:32:48

12. Després tenim que crear les vistes de les pàgines web necessàries:

```

1  eimantas, hace 4 semanas | 1 author (eimantas)
2  <!DOCTYPE html>
3  <html lang="{{ str_replace('_', '-', app()->getLocale()) }}">
4      <head>
5          <meta charset="utf-8">
6          <meta name="viewport" content="width=device-width, initial-scale=1">
7          <title>Answer</title>
8      </head>
9      <body class="antialiased">
10         <table class="default">
11             <tr>
12                 <th>Id</th>
13                 <th>Nom</th>
14                 <th>Descripció</th>
15                 <th>Usuari</th>
16                 <th>Estat</th>
17             </tr>
18             <tr>
19                 <td>{{ $form->id }}</td>
20                 <td>{{ $form->name }}</td>
21                 <td>{{ $form->description }}</td>
22                 <td>{{ $form->user_id }}</td>
23                 <td>{{ $form->active }}</td>
24             </tr>
25         </table>
26     </body>
27 </html>

```

13. Per implementar el JavaScript i utilitzarem el vue e inertia:

```

1  <template>
2      <div class="flex gap-5">
3          <div>
4              <label class="block mb-1">Min Number</label>
5              <input type="number" v-model="min" :max="max" class="bg-transparent border-white rounded-xl">
6          </div>
7
8          <div>
9              <label class="block mb-1">Max Number</label>
10             <input type="number" v-model="max" :min="min" class="bg-transparent border-white rounded-xl">
11         </div>
12
13         <div>
14             <label class="block mb-1">Default Number</label>
15             <input type="number" :value="value" :min="min" :max="max" class="bg-transparent border-white rounded-xl">
16         </div>
17     </div>
18 </template>
19
20 <script setup>
21     import { ref } from 'vue'
22
23     const props = defineProps({
24         value: Number,
25         min: Number,
26         max: Number,
27     })
28
29     const value = ref(props.value)

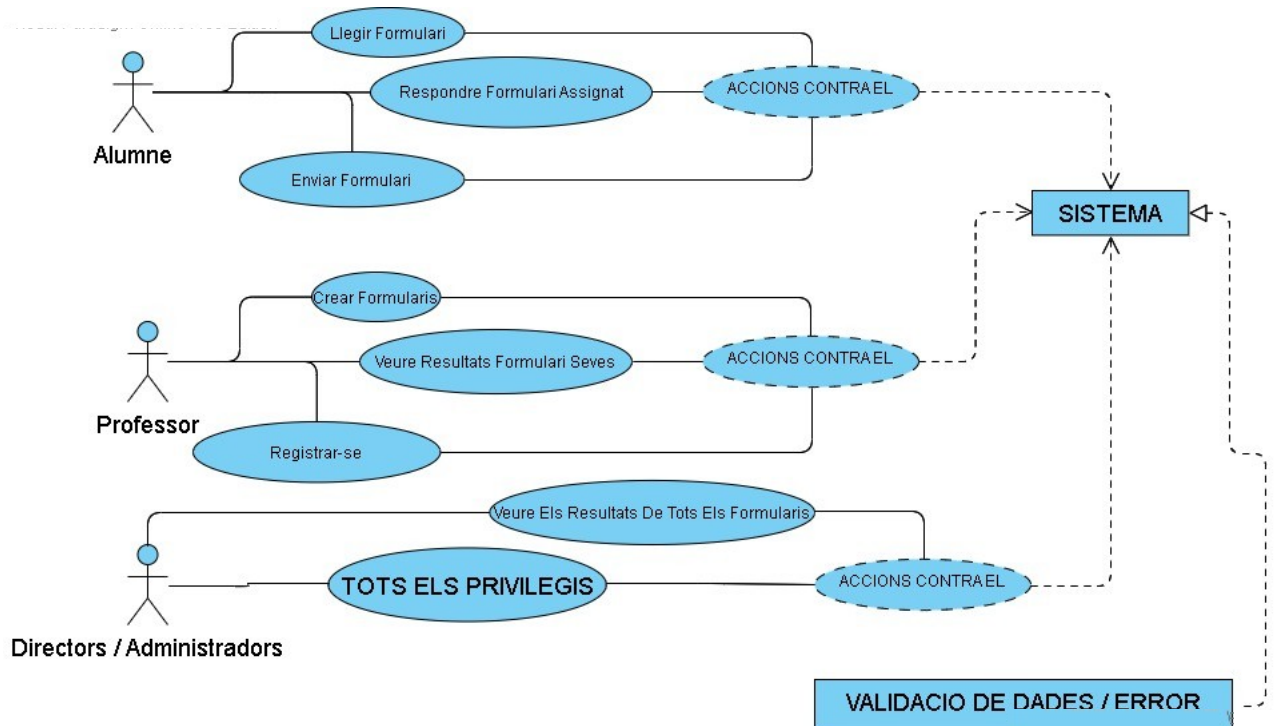
```

14. Pàgina Formulari:

The screenshot shows a web browser window with the address bar displaying '127.0.0.1:8000'. The main content area is dark-themed and contains the following elements:

- Nom del formulari**: A large text input field for the form name.
- Descripció del formulari**: A smaller text input field for the form description.
- Section Management**: A section titled 'Títol de la secció' with a dropdown arrow on the left and up/down arrows, a copy icon, and a delete icon (red X) on the right. Below this is a question management box containing:
 - Títol de la pregunta**: A text input field for the question title.
 - Text**: A dropdown menu for the question type.
 - Up/down arrows, a copy icon, and a delete icon (red X) for the question.
 - + PREGUNTA**: A button to add a new question.
- Role Management**: A section titled 'Cercar rols...' with a dropdown arrow and a plus icon. Below it are two buttons: 'Direcció ×' and 'Professor ×'.
- + SECCIÓ**: A button to add a new section.
- Crear formulari**: A green button to create the form.

Diagrama De Casos D'ús



- **Escenari:**

1. L'usuari accedeix a la pàgina web de CendraForms
2. L'usuari rol convidat pot navegar per la pàgina
3. Iniciar Sessió / Registrar: oAuth.
4. Un cop registrat, automàticament l'usuari té el rol de: Alumne .
5. Rols existents assignat de manera manual per l'administrador: Professor / Admin / Director .
6. Si les credencials son correctes, l'usuari pot accedir al seu compte privat.
7. L'usuari ara pot fer les següents accions a la pàgina web:

- Veure el Formulari
- Respondre Formulari
- Enviar Formulari

8. L'usuari amb el rol: Professor, pot crear formularis i veure els seus formularis.

9. L'usuari amb el rol: Director / Admin, por veure tots els formularis de tots els professors.

- **Actors / Rols:**

- En el nostre sistema tenim 3 tipus de rols:

- **ADMINISTRADOR / DIRECTOR:**

- **Requeriments:**

Nom	Administrador / Director
Autor	Nom del Administrador / Director
Data	Data quant és registra el Administrador / Director en el sistema
Descripció	Té el rol de admin / director. Administra tot el sistema.
Actors	Ocupa la categoria alta en privilegis
Precondicions	L'administrador / director té que estar autenticat en el sistema.

- **PROFESSOR:**

- **Requeriments:**

Nom	PROFESSOR
Autor	Nom del Professor
Data	Data quant és registra el Professor en el sistema
Descripció	Té el rol de Professor. Podrà afegir, editar i esborrar cursos (softdelete). Tindrà tots els permisos dels usuaris.
Actors	Ocupa la categoria mitjana en privilegis
Precondicions	El Professor té que estar autenticat en el sistema.

- **ALUMNE:**

- **Requeriments:**

Nom	ALUMNE
Autor	Nom del Alumne
Data	Data quant és registra el Alumne en el sistema
Descripció	Té el rol de Alumne. Estudiar cursos, guardar-los com favorits i reportar errors.
Actors	Ocupa la categoria baixa en privilegis
Precondicions	L'usuari té que estar autenticat en el sistema.

- **Requeriments del sistema:**

- El servidor ha de permetre als usuaris accedir, registrar, iniciar sessió i veure els formularis, segons el rol que tingui cada usuari tindrà uns privilegis o altres per fer determinades accions.