UKAhoot!

Authors:
Aarón Arias Pérez
José Crespo Guerrero

Contents

1. Objectives
2. Tools
3. Ukahoot!
   3.1. Frontend
   3.2. Backend
4. Security measures
5. Demonstration

1. Objectives

Objectives

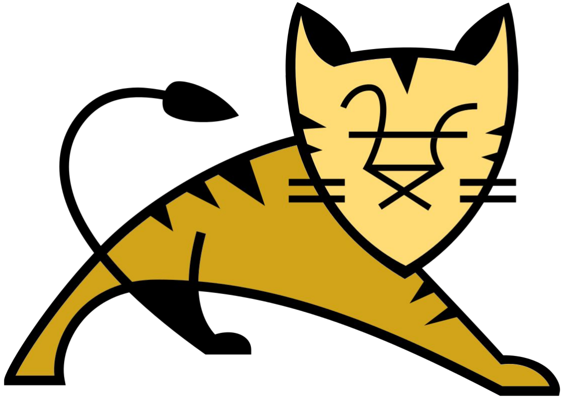
- Create a Rest API
- Create a web application
- Secure application and api

2. Tools

ionic

jQuery
write less, do more.

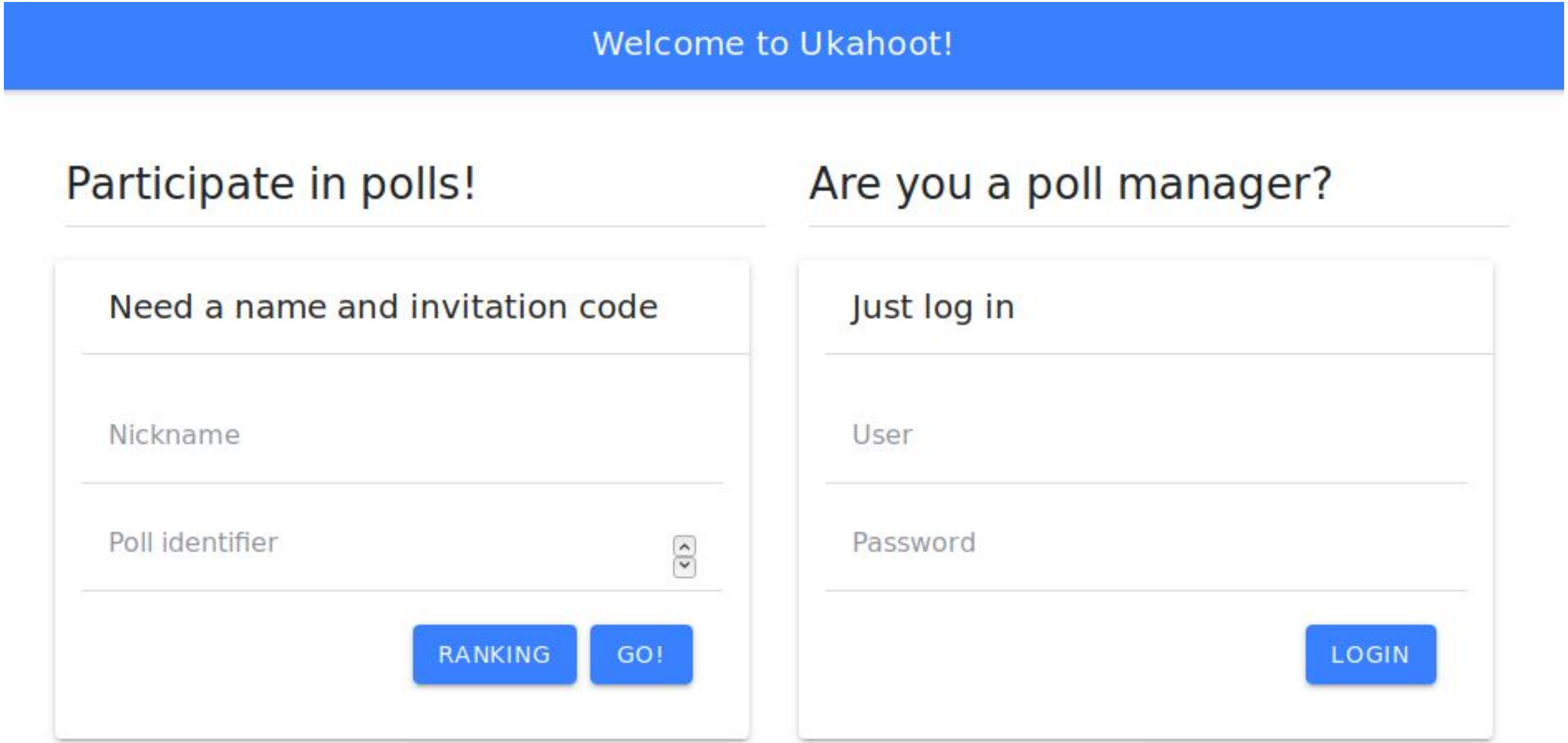
AJAX
Asynchronous Javascript And XML



3. Ukahoot!

UKAhoot!

Frontend tour - Index



Welcome to Ukahoot!

Participate in polls!

Need a name and invitation code

Nickname

Poll identifier

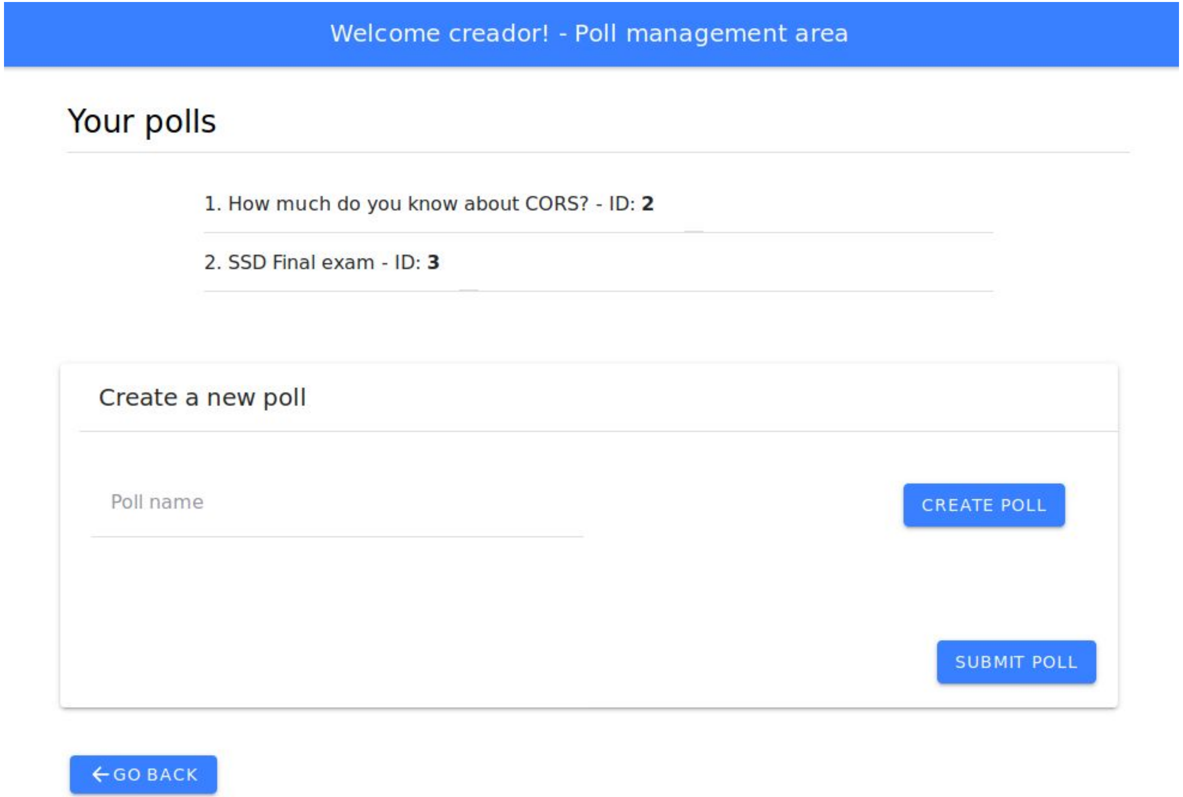
Are you a poll manager?

Just log in

User

Password

Frontend tour - Dashboard



Welcome creador! - Poll management area

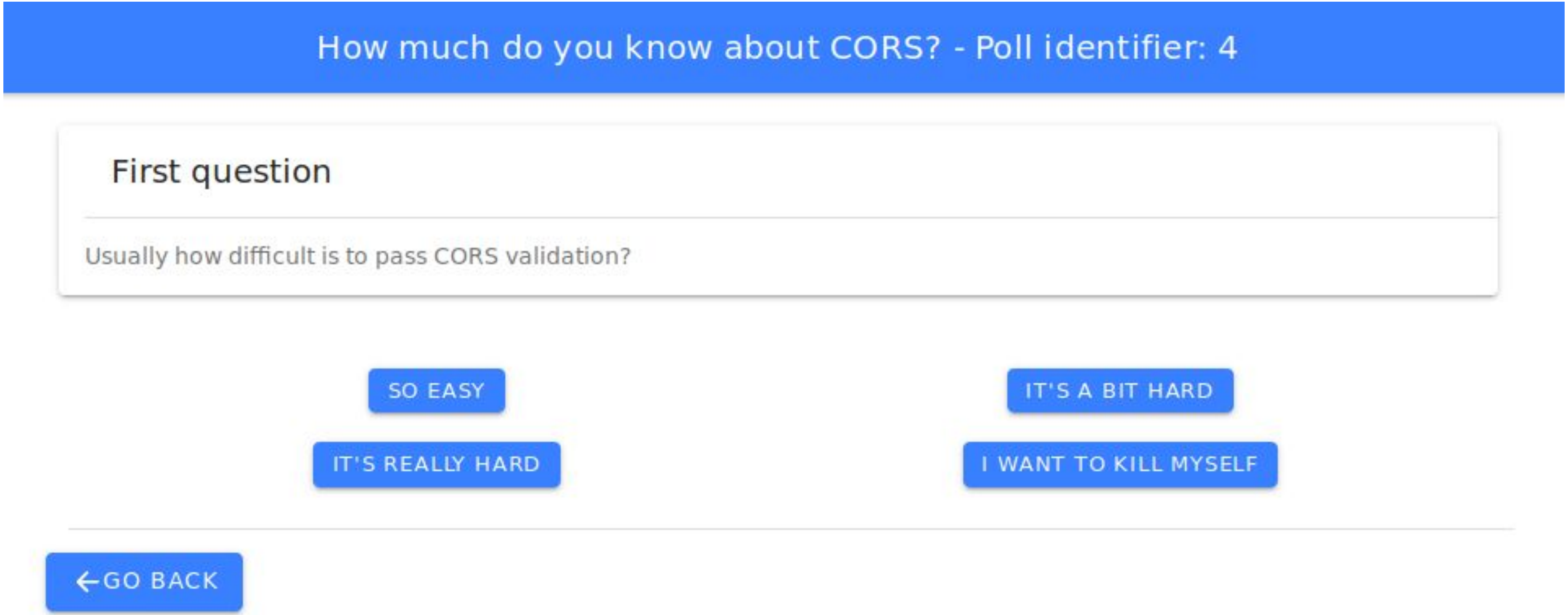
Your polls

1. How much do you know about CORS? - ID: **2**
2. SSD Final exam - ID: **3**

Create a new poll

Poll name

Frontend tour - Poll questions

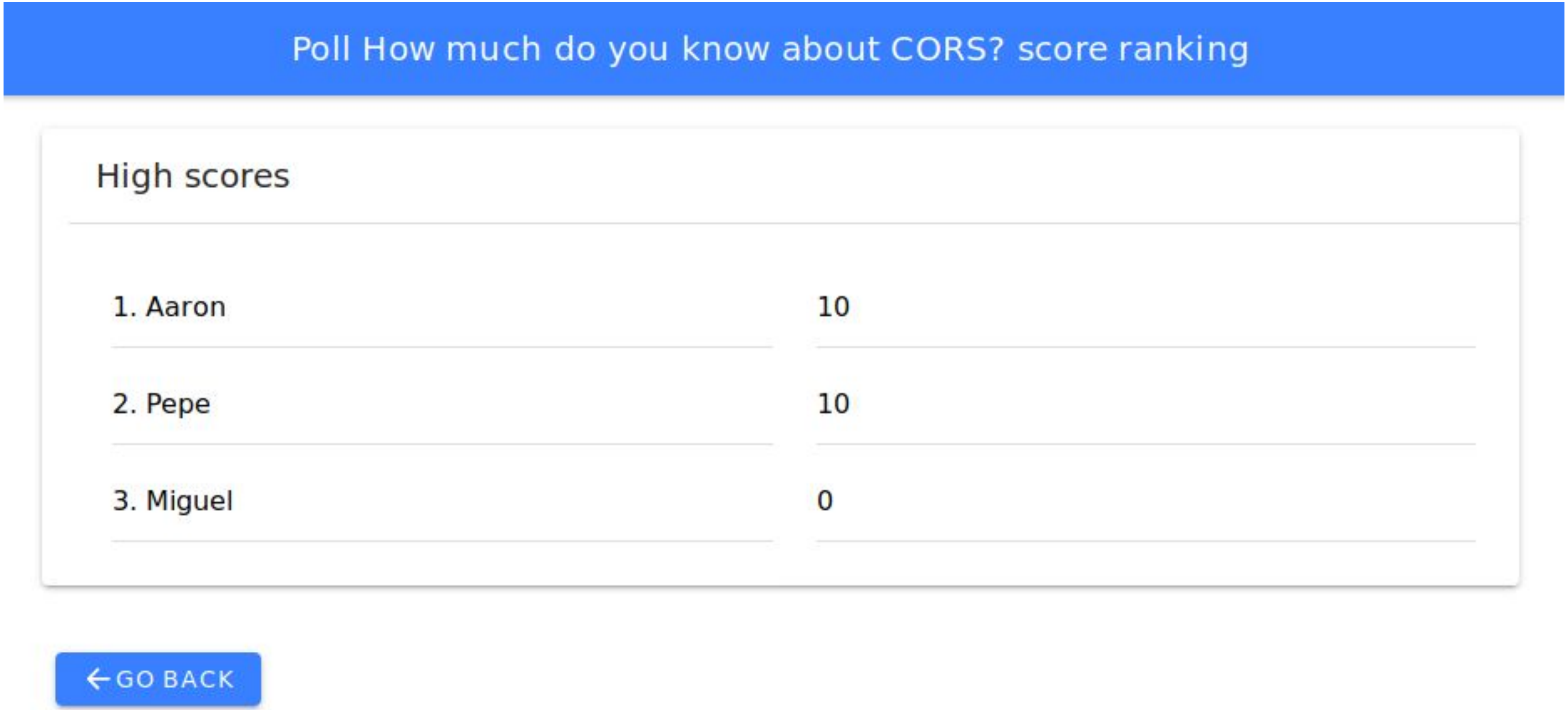


How much do you know about CORS? - Poll identifier: 4

First question

Usually how difficult is to pass CORS validation?

Frontend tour - Ranking



Poll How much do you know about CORS? score ranking

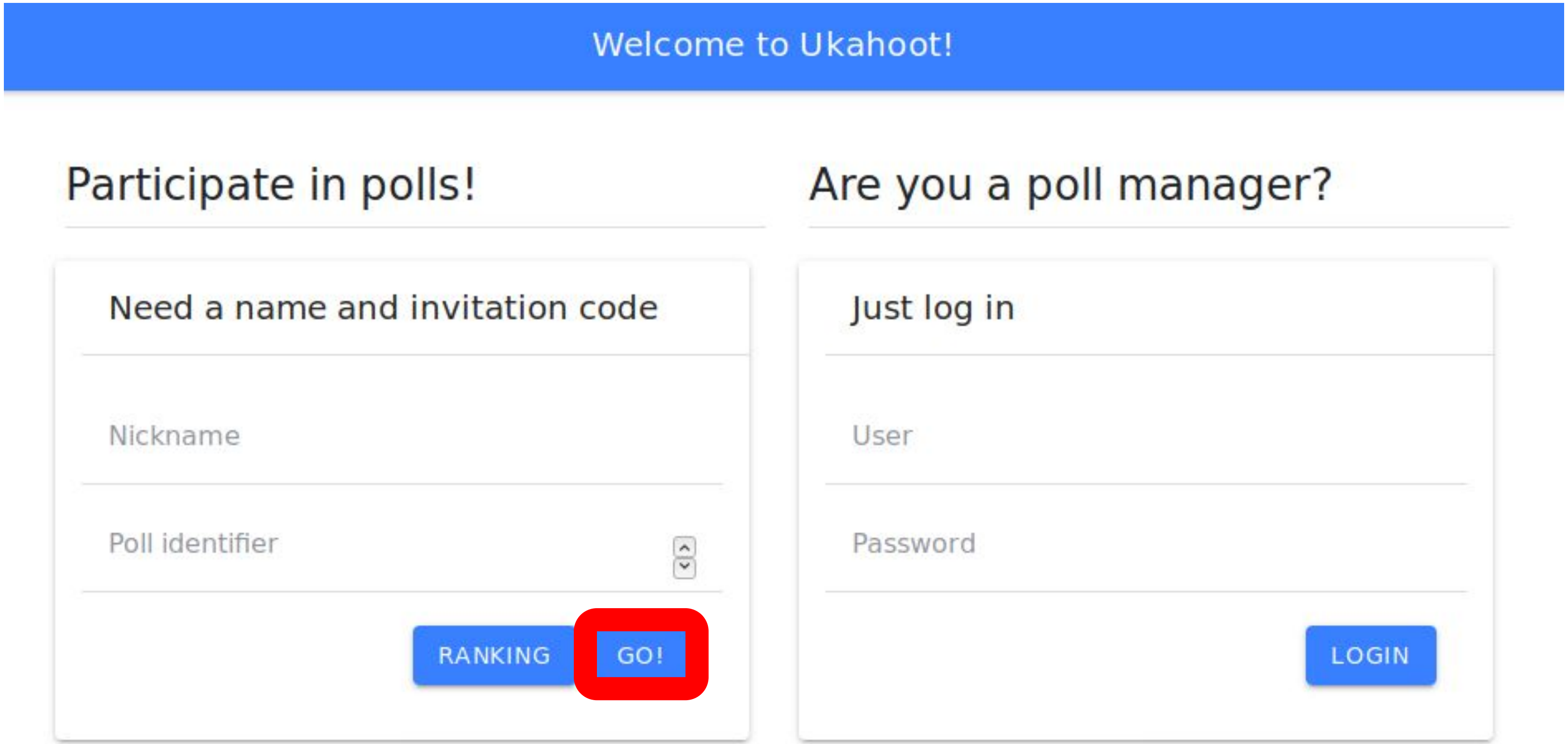
High scores

1. Aaron	10
2. Pepe	10
3. Miguel	0

Backend

- /getApikey
- /play
- /enviar_respuestas
- /auth
- /encuesta
- /ranking/{id}

/getApikey



Welcome to Ukahoot!

Participate in polls!

Need a name and invitation code

Are you a poll manager?

Just log in

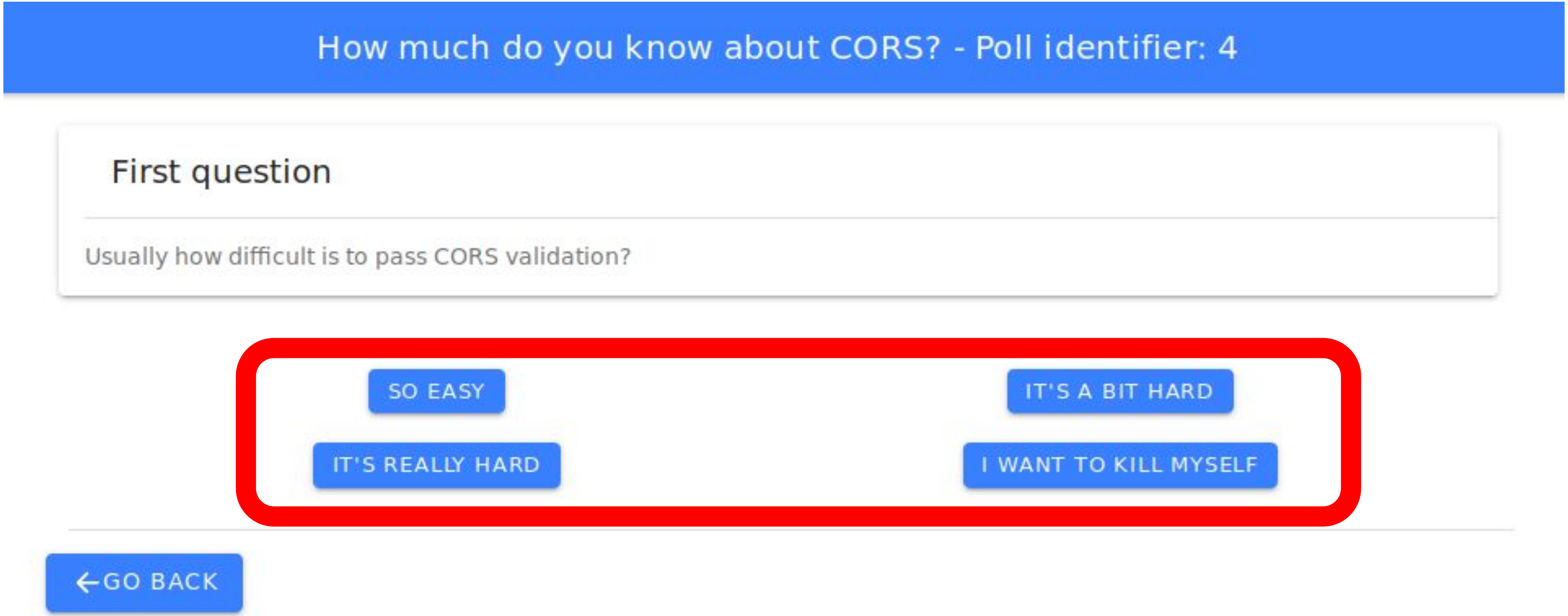
/getApikey

- HTTP method: POST
- Input: Nickname and poll identifier
- Output: Apikey
- Generate and returns an apikey

/play

- HTTP method: POST
- Input: Nickname, poll identifier and apikey
- Output: Poll name and questions
- Returns the poll with its questions

/enviar_respuestas



How much do you know about CORS? - Poll identifier: 4

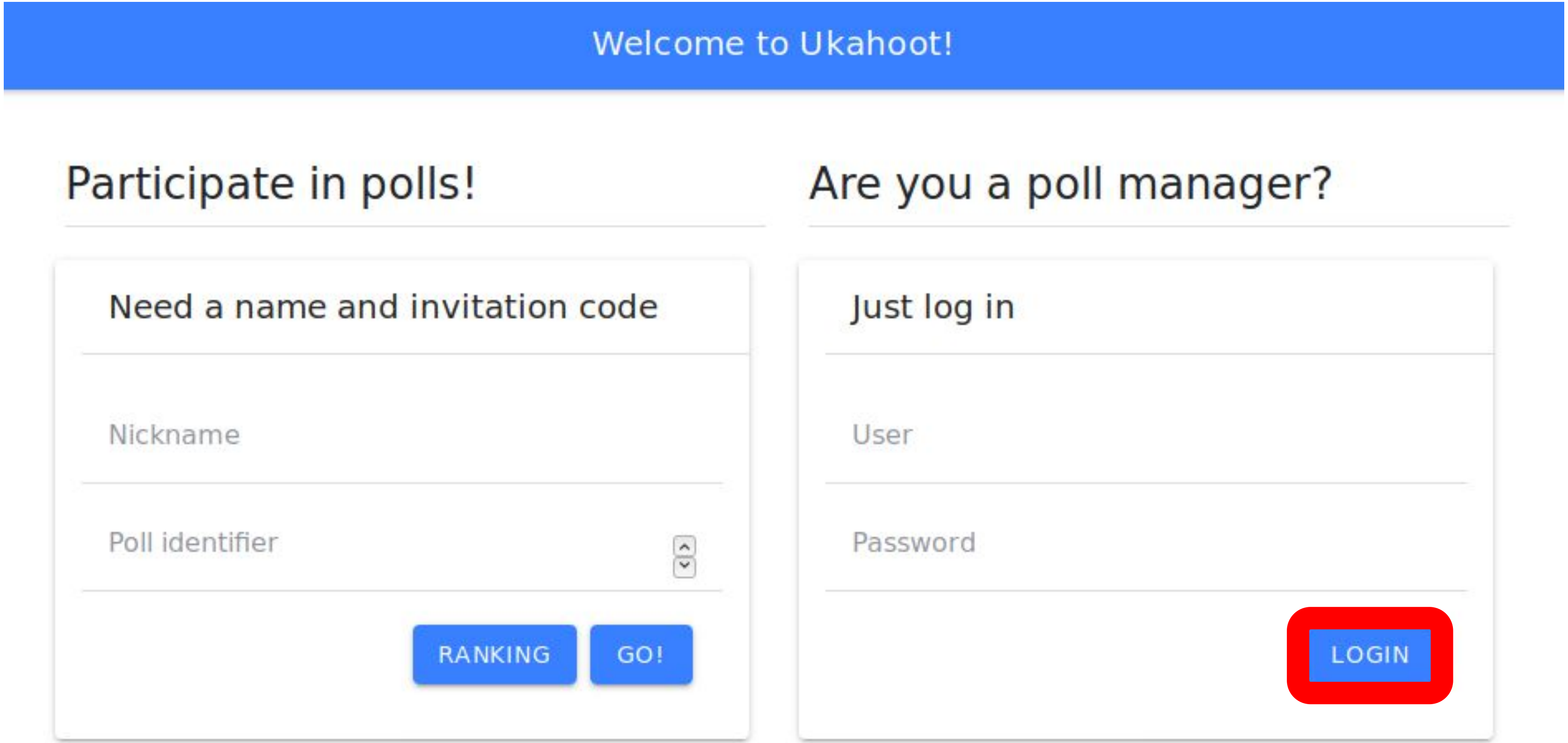
First question

Usually how difficult is to pass CORS validation?

/enviar_respuestas

- HTTP method: POST
- Input: Answers, poll identifier, apikey, nickname
- Output: Message
- Compute the score obtained and write it in the ranking table and returns a message

/auth



Welcome to Ukahoot!

Participate in polls!

Need a name and invitation code

Nickname

Poll identifier

Are you a poll manager?

Just log in

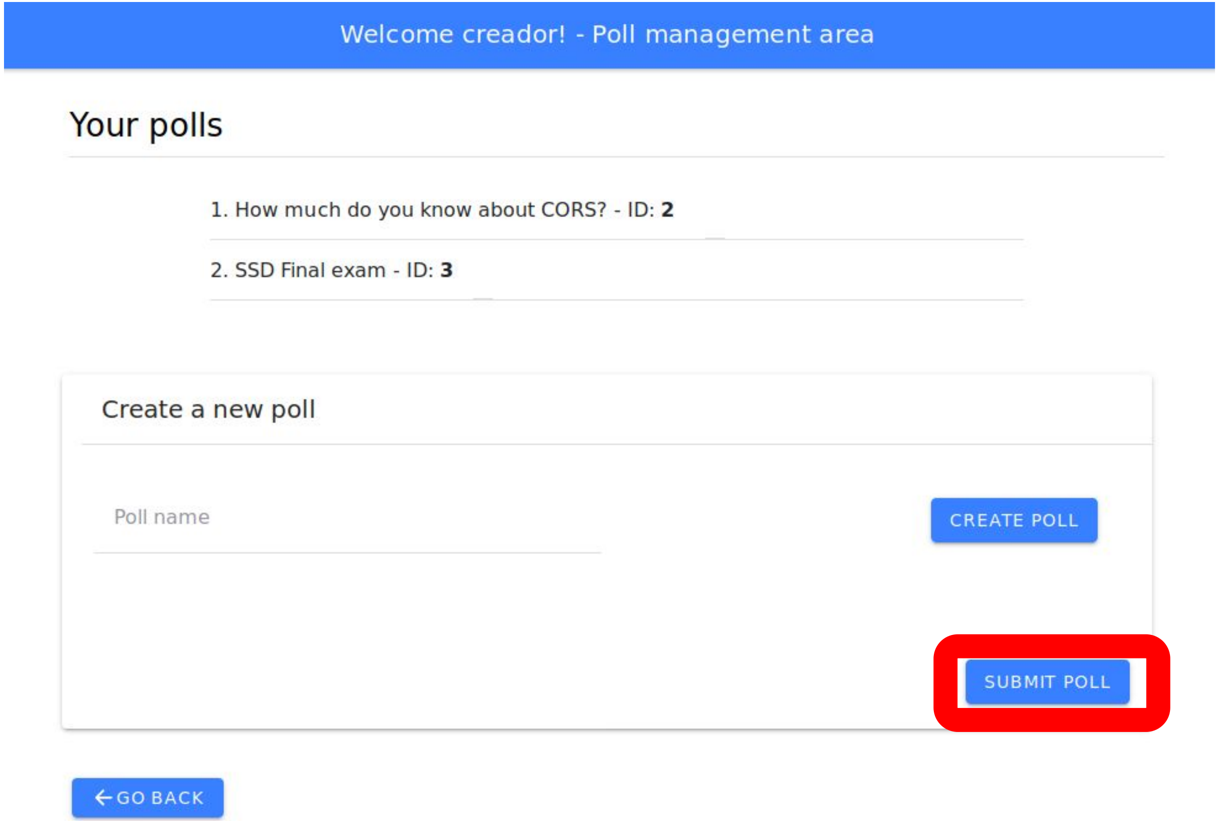
User

Password

/auth

- HTTP method: POST
- Input: Username and password
- Output: JWT token and message
- Validates user and password, generates and returns a token

/encuesta



Welcome creador! - Poll management area

Your polls

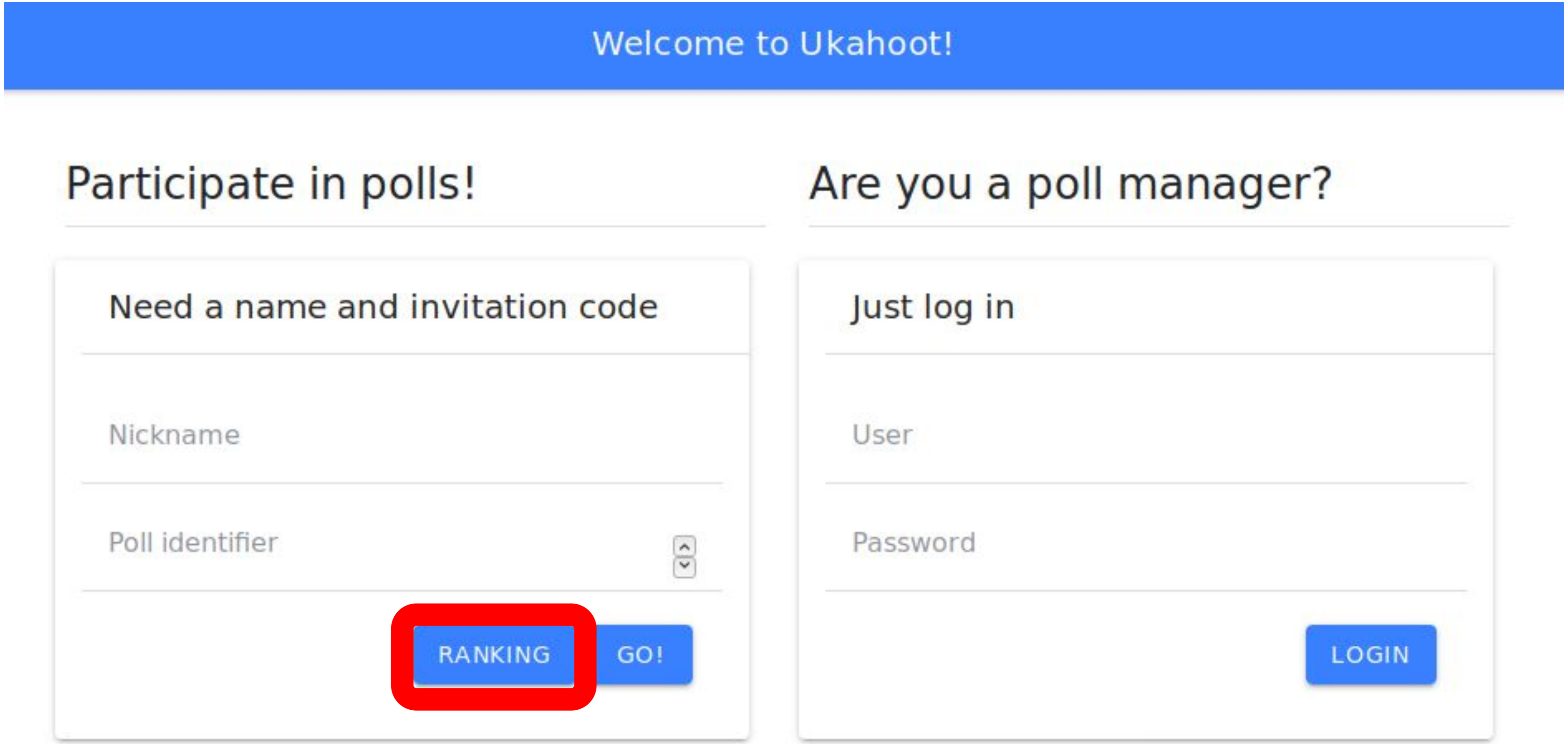
1. How much do you know about CORS? - ID: **2**
2. SSD Final exam - ID: **3**

Create a new poll

/encuesta

- HTTP method: POST
- Input: Token, username and poll
- Output: Poll identifier
- Creates the poll and the ranking table

/ranking/{id}



Welcome to Ukahoot!

Participate in polls!

Need a name and invitation code

Nickname

Poll identifier

Are you a poll manager?

Just log in

User

Password

/ranking/{id}

- HTTP method: GET
- Input: Poll identifier
- Output: Ranking
- Returns a ranking table of the selected poll

Frontend properties

- jQuery + ionic framework
- Input validation
- Dynamic loading with Ajax
- Session cookies
- HTTPS requests to API REST

4. Security measures

Feature	Value
HTTPS	0,3
Access Control	0,6
JWT	1,3
Api Keys	0,6
Restric HTTP Methods	0,3
Input Validation	0,9
Validate Content Types	0,3
Management endpoint	0,3
Error handling	0,6
Audit logs	0,3
Security headers	0,3
CORS	1,6
Sensitive Information in HTTP requests	0,3
HTTP return code	0,3

Https

- Generate a local certificate with Java’s keytool
- Setup Tomcat to:
    - SSL encrypt
    - Expose HTTPS port
    - Use local certificate and keystore password
- Configuration found in server.xml

Feature	Value
HTTPS	0,3
Access Control	0,6
JWT	1,3
Api Keys	0,6
Restric HTTP Methods	0,3
Input Validation	0,9
Validate Content Types	0,3
Management endpoint	0,3
Error handling	0,6
Audit logs	0,3
Security headers	0,3
CORS	1,6
Sensitive Information in HTTP requests	0,3
HTTP return code	0,3

Access control

- We have used a token (api key or jwt) to maintain a session with the help of a cookie
- The client application manages the cookie
- The backend have the registered users in memory
- Tokens are linked to users

Feature	Value
HTTPS	0,3
Access Control	0,6
JWT	1,3
Api Keys	0,6
Restric HTTP Methods	0,3
Input Validation	0,9
Validate Content Types	0,3
Management endpoint	0,3
Error handling	0,6
Audit logs	0,3
Security headers	0,3
CORS	1,6
Sensitive Information in HTTP requests	0,3
HTTP return code	0,3

Jwt

- The backend generates the jwt token and it also validates it when it is necessary
- The client has the token saved in cookies and it manages them to keep the session
- It is only applied for poll creators

Feature	Value
HTTPS	0,3
Access Control	0,6
JWT	1,3
Api Keys	0,6
Restric HTTP Methods	0,3
Input Validation	0,9
Validate Content Types	0,3
Management endpoint	0,3
Error handling	0,6
Audit logs	0,3
Security headers	0,3
CORS	1,6
Sensitive Information in HTTP requests	0,3
HTTP return code	0,3

Api keys

- The backend generates the api key and it also validates it when it is necessary
- The client has the token saved in cookies and it manages them to keep the session
- It is only applied for participants

Feature	Value
HTTPS	0,3
Access Control	0,6
JWT	1,3
Api Keys	0,6
Restric HTTP Methods	0,3
Input Validation	0,9
Validate Content Types	0,3
Management endpoint	0,3
Error handling	0,6
Audit logs	0,3
Security headers	0,3
CORS	1,6
Sensitive Information in HTTP requests	0,3
HTTP return code	0,3

Restrict http methods

- Paths serves only on necessary methods
- Methods restrictions in web.xml
- Apikey token or JWT is mandatory for POST requests
- GET requests don’t need a token

Feature	Value
HTTPS	0,3
Access Control	0,6
JWT	1,3
Api Keys	0,6
Restric HTTP Methods	0,3
Input Validation	0,9
Validate Content Types	0,3
Management endpoint	0,3
Error handling	0,6
Audit logs	0,3
Security headers	0,3
CORS	1,6
Sensitive Information in HTTP requests	0,3
HTTP return code	0,3

Input validation

- At client side:
  - Native input fields type
  - Basic JS validation
- At server side:
  - Types validation

Feature	Value
HTTPS	0,3
Access Control	0,6
JWT	1,3
Api Keys	0,6
Restric HTTP Methods	0,3
Input Validation	0,9
Validate Content Types	0,3
Management endpoint	0,3
Error handling	0,6
Audit logs	0,3
Security headers	0,3
CORS	1,6
Sensitive Information in HTTP requests	0,3
HTTP return code	0,3

Validate content types

- It applies to the request data types (headers or body)
- The content-types are validated implicitly by the api rest core

Feature	Value
HTTPS	0,3
Access Control	0,6
JWT	1,3
Api Keys	0,6
Restric HTTP Methods	0,3
Input Validation	0,9
Validate Content Types	0,3
Management endpoint	0,3
Error handling	0,6
Audit logs	0,3
Security headers	0,3
CORS	1,6
Sensitive Information in HTTP requests	0,3
HTTP return code	0,3

Management endpoint

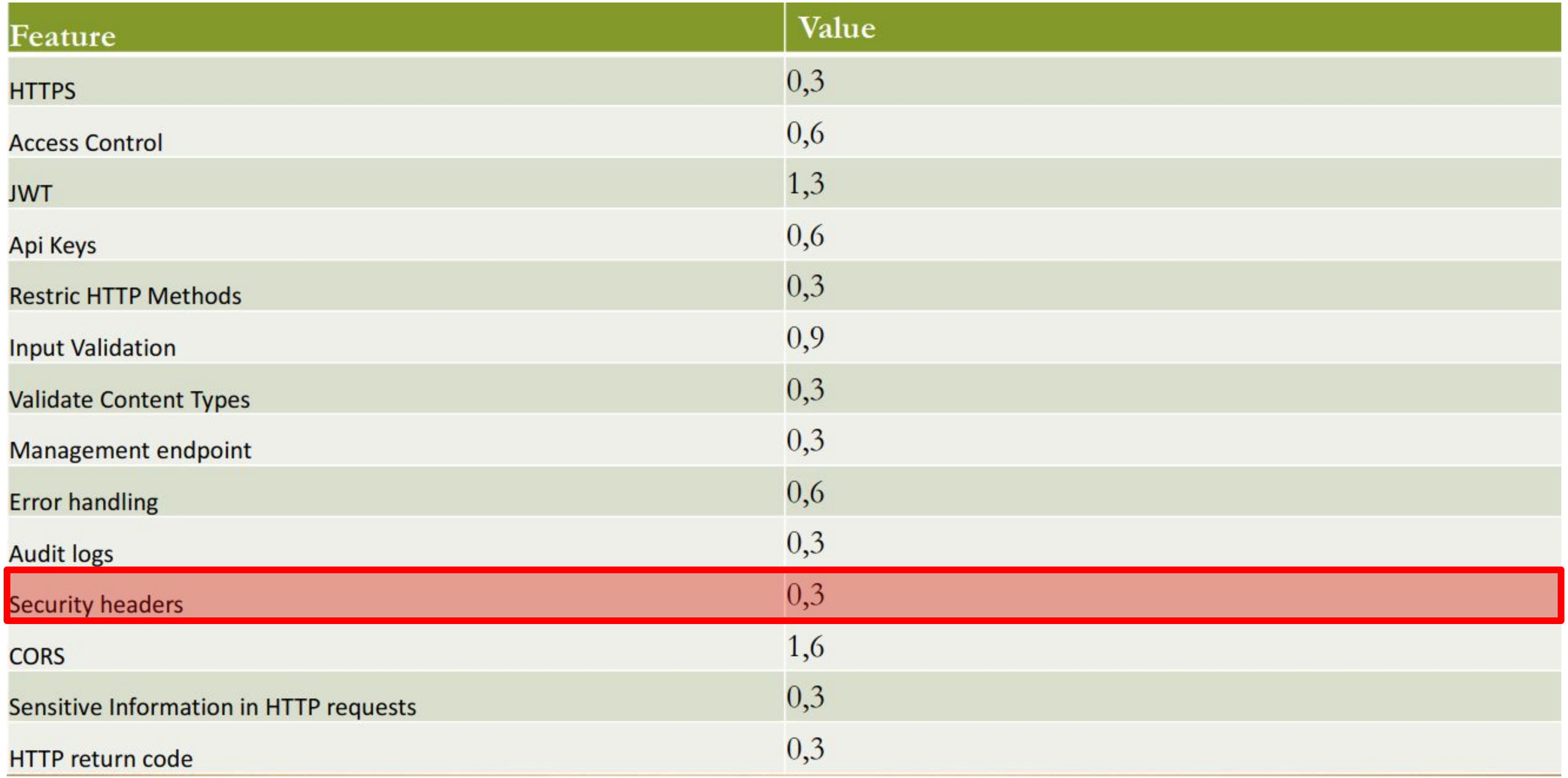
- Login secured management endpoints
- JWT token for authentication
- Session cookies expiry

Feature	Value
HTTPS	0,3
Access Control	0,6
JWT	1,3
Api Keys	0,6
Restric HTTP Methods	0,3
Input Validation	0,9
Validate Content Types	0,3
Management endpoint	0,3
Error handling	0,6
Audit logs	0,3
Security headers	0,3
CORS	1,6
Sensitive Information in HTTP requests	0,3
HTTP return code	0,3

Error handling

- Error handling at the Client level (ajax error management)
- Error handling at the api level (exceptions)

Feature	Value
HTTPS	0,3
Access Control	0,6
JWT	1,3
Api Keys	0,6
Restric HTTP Methods	0,3
Input Validation	0,9
Validate Content Types	0,3
Management endpoint	0,3
Error handling	0,6
Audit logs	0,3
Security headers	0,3
CORS	1,6
Sensitive Information in HTTP requests	0,3
HTTP return code	0,3



Feature	Value
HTTPS	0,3
Access Control	0,6
JWT	1,3
Api Keys	0,6
Restric HTTP Methods	0,3
Input Validation	0,9
Validate Content Types	0,3
Management endpoint	0,3
Error handling	0,6
Audit logs	0,3
Security headers	0,3
CORS	1,6
Sensitive Information in HTTP requests	0,3
HTTP return code	0,3

Feature	Value
HTTPS	0,3
Access Control	0,6
JWT	1,3
Api Keys	0,6
Restric HTTP Methods	0,3
Input Validation	0,9
Validate Content Types	0,3
Management endpoint	0,3
Error handling	0,6
Audit logs	0,3
Security headers	0,3
CORS	1,6
Sensitive Information in HTTP requests	0,3
HTTP return code	0,3

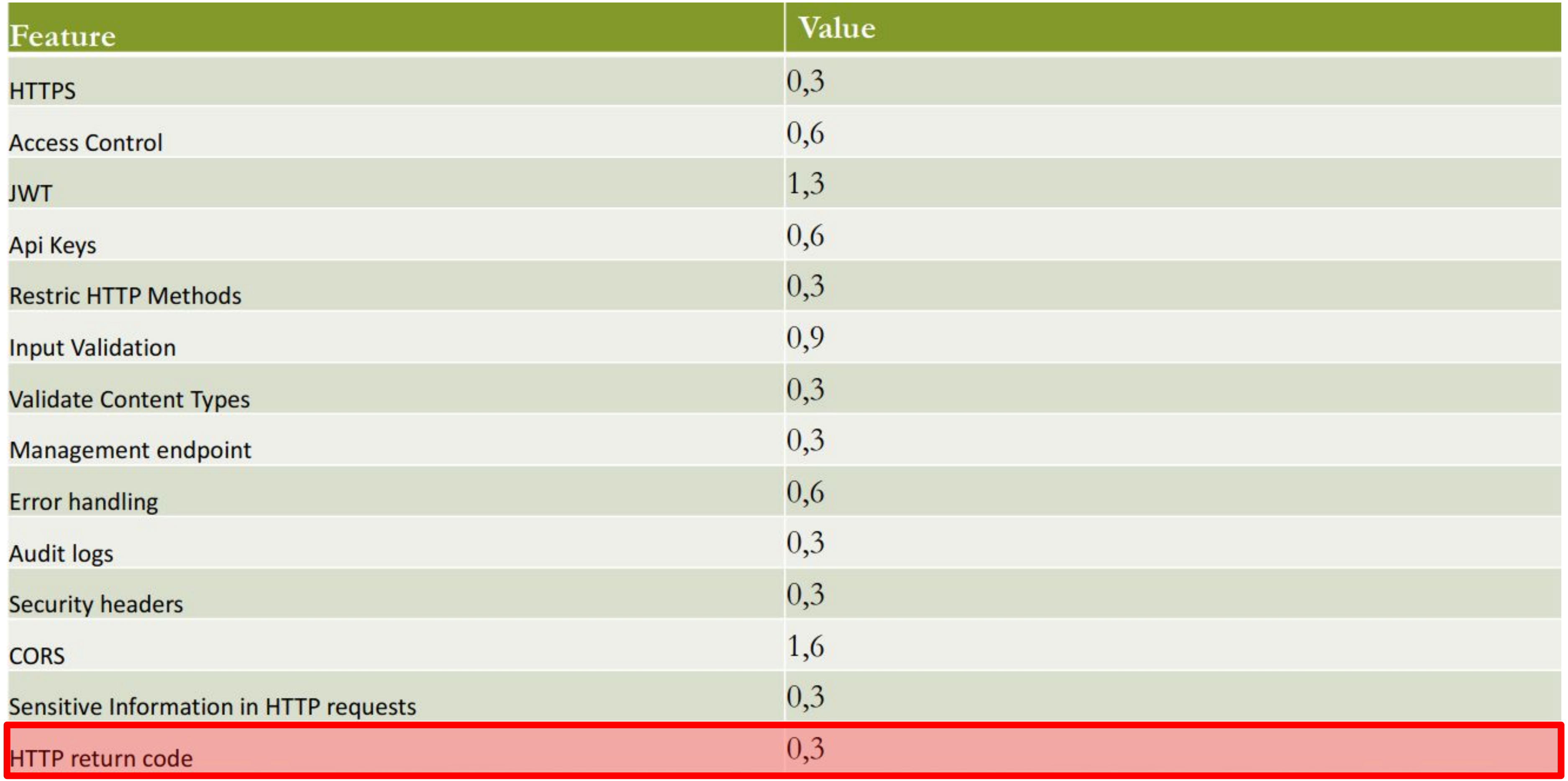
CORS

- Implemented CORS filter
  - Each response is filled with additional headers
  - Access-Control-Allow-Origin
  - Access-Control-Allow-Methods
  - Access-Control-Allow-Headers
  - Access-Control-Allow-Credentials
- Setup of web.xml
  - Match url with CORS filter
- crossDomain: true at client-side requests

Feature	Value
HTTPS	0,3
Access Control	0,6
JWT	1,3
Api Keys	0,6
Restric HTTP Methods	0,3
Input Validation	0,9
Validate Content Types	0,3
Management endpoint	0,3
Error handling	0,6
Audit logs	0,3
Security headers	0,3
CORS	1,6
Sensitive Information in HTTP requests	0,3
HTTP return code	0,3

Sensitive information in http requests

- The sensitive information (usernames, passwords, …) is protected by using HTTPS



Feature	Value
HTTPS	0,3
Access Control	0,6
JWT	1,3
Api Keys	0,6
Restric HTTP Methods	0,3
Input Validation	0,9
Validate Content Types	0,3
Management endpoint	0,3
Error handling	0,6
Audit logs	0,3
Security headers	0,3
CORS	1,6
Sensitive Information in HTTP requests	0,3
HTTP return code	0,3

5. Demonstration

Questions?