

# Cloud Applications

Teach Me

September 21, 2016

Opgesteld door: Rival Thompson & Hans Van Staey

# Cloud Applications

---

Teach Me

## Project omschrijving

- *Probleemstelling*

Vele leerkrachten willen meer interactiviteit tijdens de lessen, van meer dan enkel een klein aantal leerlingen. Het probleem is dat de antwoorden van studenten beïnvloed worden door andere studenten die vaak antwoord geven op de vragen die worden gesteld. Hiervoor wil de docent gebruik maken van een digitaal platform waarop hij of zij de studenten de kans wil bieden om met hun eigen antwoord de gestelde vragen te beantwoorden. De docent verwacht dat het een open concept is dit wil zeggen dat er meerdere soorten vragen kunnen zijn, bijvoorbeeld de open vraag, gesloten vraag, code, enzovoort...

- *Doel van het project*

Het project heeft als doel een digitaal leerplatform voorzien waar lectoren en studenten interactief kunnen bezig zijn in de les. Zodat de docent zijn of haar vragen kan stellen aan al zijn studenten zonder dat er een restrictie is op de vragen of de persoonlijkheid van de antwoorden van de studenten. Dit alles met een zo eenvoudig mogelijke en gestroomlijnde gebruikerservaring voor zowel de studenten als de docenten.

- *Methodologie*

We gaan te werk volgens het scrum principe, dit is een flexibele manier om een softwareproduct te ontwikkelen. We werken elk als een multidisciplinair team in korte sprints. Onze sprints duren steeds twee weken, in deze tijd leveren we telkens een uitbreiding op onze werkende toepassing. Deze scrum techniek gaan we gebruiken omdat de klant gaandeweg nog aanpassingen kan vragen en deze zonder al te veel moeilijkheden verwerkt moeten worden. Per sprint gaan we ons baseren op een aantal user stories, die in onze backlog of werkvoorraad staan, om onze volgende taak uit te voeren. Deze stories gaan van belangrijkst naar minst nuttig. De stories die we dan afnemen verdelen we op team sterkte, het beste team krijgt de voorkeur om die story uit te werken, maar als het andere team dit ook wil doen kan nog overlegd worden wie de story krijgt.

Als er zich problemen voordoen, overleggen we die zoals tijdens een daily scrum meeting. Aan het einde van elke sprint gaan we het resultaat van de afgelopen sprint evalueren, dit doen we om er zeker van te zijn dat elke story afgewerkt is, dit zal de sprint review zijn. Daarna wordt de sprint planning gemaakt voor de volgende reeks stories.

Omdat we met Meteor gaan werken wat een nieuw platform voor ons beide is, moeten we ons hierin nog verdiepen. Dat gaan we tijdens onze 'spike' doen,

wat een korte periode is om onderzoek te doen of hoe we met Meteor moeten werken en ook wat de voor- en nadelen hiervan gaan zijn. Deze spike zal plaatsvinden voor onze eerste sprint.

Onze productowner is meneer Peeters, wij zijn het ontwikkelteam en scrummaster, samen vormen deze het scrumteam.

- *Verwachte resultaten*

Het gewenste resultaat is een webapp waarbij de docent zich kan aanmelden en een les aanmaakt. Hierna kan hij instellen of de antwoorden van de studenten anoniem moeten blijven of niet. Hier kan hij al vragen aanmaken, of deze on the fly aanmaken tijdens de les.

Eenmaal de docent klaar is om zijn les te starten wordt er een token gegenereerd waarmee de studenten kunnen deelnemen aan zijn of haar les. Op het moment dat de lector besluit een vraag te stellen aan zijn of haar studenten verschijnt deze vraag op het scherm van de studenten. Dit kan een open of multiple choice vraag zijn, maar ook een teken opdracht of code voorbeeld is mogelijk. Indien het een code voorbeeld is kan de docent op deze code klikken en wordt deze gekopieerd naar zijn klembord, daar kan de docent deze dan plakken in zijn of haar code IDE voor het testen.

Via een feedback rating kan een de student meedelen of hij of zij de vraag begrijpt of nog extra uitleg zou willen. Eenmaal de docent besluit dat er genoeg tijd is geweest om de vraag te beantwoorden sluit hij de vraag af. Afhankelijk van het type vraag worden de meest voorkomende of alle antwoorden of percentages, etc. weergegeven op het scherm van de docent. Waarbij hij of zij de antwoorden kan overlopen en eventuele verbeteringen kan aanbrengen. Aan het einde van een les kan de docent besluiten de les te bewaren voor later of alles te verwijderen. Als de lector voorgaande keuze heeft gemaakt wordt hij of zij teruggeleid naar het scherm waar een volgende les aangemaakt kan worden.

## Technologie

Enkele van de technologieën waarvan we weten dat we ze gaan gebruiken zijn:

Meteor: Dit is een javascript web framework dat zeer goed is voor een webapp die actief moet werken met data. Het heeft ook native functionaliteit voor het communiceren met onze database en het gebruik van websockets.

Mongo DB: Onze database zal een Mongo Db database zijn die de data van de docenten, lessen, vragen en de antwoorden van de studenten zal omvatten. Deze werkt ook probleemloos samen met Meteor voor een goedwerkende en gestroomlijnde ervaring.

OAUTH: Oauth is een manier om gebruikers te authenticeren en te laten inloggen. Dit zou een geweldige surplus zijn om zo het inlog en sign-up gebeuren te versimpelen voor onze gebruikers.

## Uitkomst voor de maatschappij

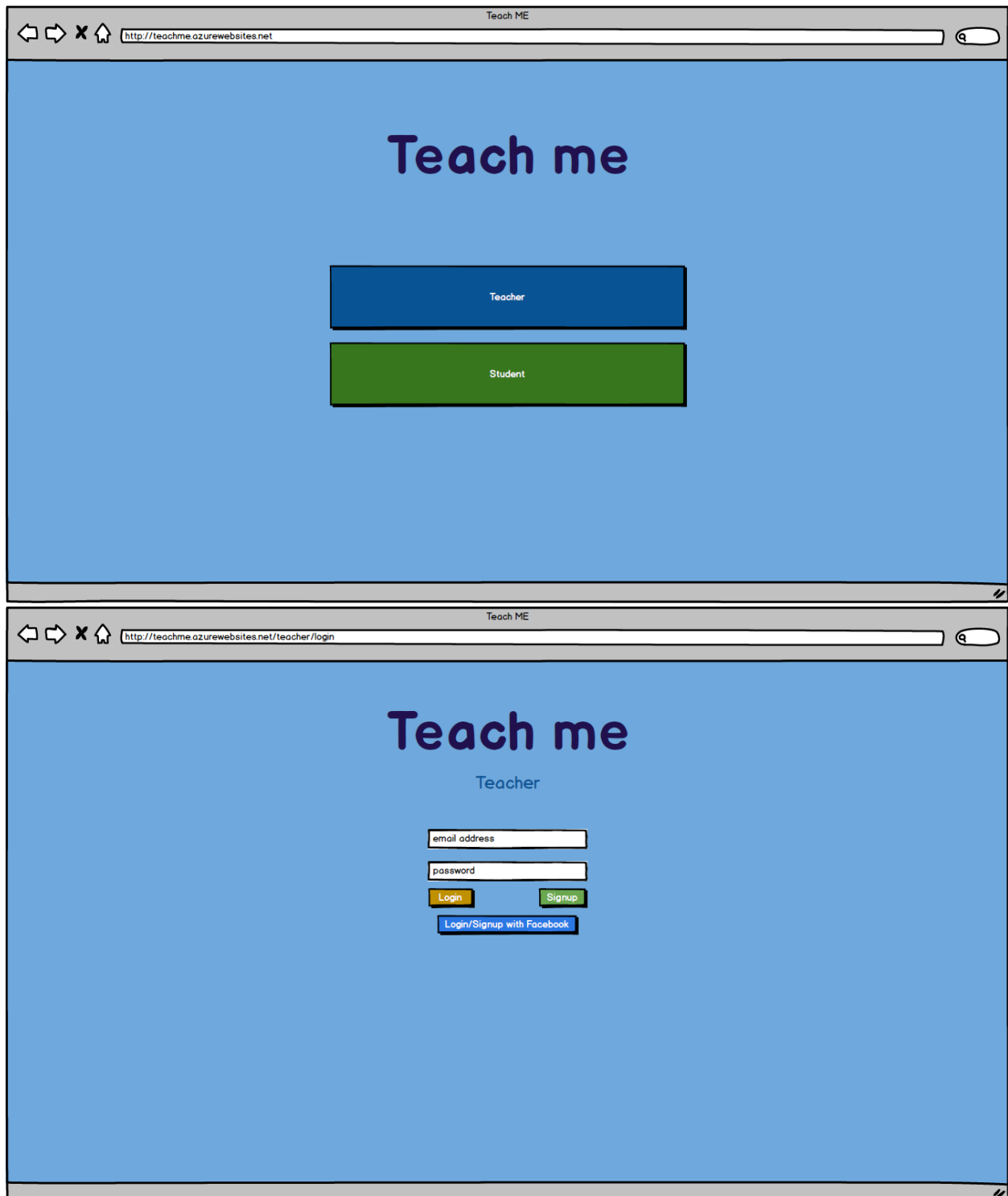
We zouden ons platform willen deployen in het onderwijs, allereerst in het hoger onderwijs. Daar zouden we minder barrières moeten neerhalen, zoals het gebruik van een laptop, tablet of smartphone. Elke student heeft wel zo'n apparaat bij zich tijdens de les, ze kunnen dit gebruiken om zo aan de les deel te nemen, wat de nood aan extra hardware voorziening weghaalt. Nu men in een deel van het lager en middelbaar onderwijs ook al tablets begint te gebruiken, kunnen we hier ook een doelpubliek in vinden.

Ons platform geeft meerdere leerlingen de kans om te kunnen antwoorden. Dit is zeker voordelig voor de leerlingen die wel het antwoord weten, maar niet graag hun hand willen opsteken of niet graag alle aandacht krijgen tijdens de les en schrik zouden hebben om fout te antwoorden. Terwijl er een student is die zo goed als altijd het correcte antwoord weet.

## User stories/ Actoren

Als een docent wil ik aanmelden & inloggen
Als een docent wil ik een les kunnen aanmaken
Als een student wil ik deel kunnen nemen aan de les
Als een docent wil ik multiple choice vragen kunnen stellen
Als een docent wil ik open vragen kunnen stellen
Als een docent wil ik vragen stellen over code
Als een docent wil ik weten wie welk antwoord gaf.
Als beheerder wil ik een database hebben waarop alles opgeslagen wordt
Als beheerder wil ik een onlineservice hebben waarop de webapp gehost wordt
Als student wil ik kunnen doorgeven aan de docent dat het niveau van de vragen te hoog is
Als docent wil ik de antwoorden kunnen tonen en overlopen met mijn studenten
Als gebruiker wil ik dat de userinterface eenvoudig en intuïtief is
Als gebruiker wil ik dat de gebruikte data beperkt blijft
Als een docent wil ik aan het einde van mijn les mijn les kunnen opslaan
Als een developer moet ik onderzoek doen naar Meteor

## Mockups



Teach ME
http://teachme.azurewebsites.net/teacher/signup

# Teach me

Teacher

email address

password

repeat password

first name

last name

school name and address

Signup

Teach ME
http://teachme.azurewebsites.net/teacher/dashboard

Teach me
Teacher
Change Password
Logout
Start Lesson

## Lesson1

Lesson1

Lesson2

Lesson3

Lesson4

Lesson5

Lesson6

Lesson7

Question1

Answer1

Answer2

Answer3

...

Question2

Answer1

Answer2

Answer3

...

Question3

...

## New Lesson

Lesson Name:

Lesson Subject:

☒ Student anonymity

Initial Questions

question1

Open

question2

Multiple Choice

Option1

Option2

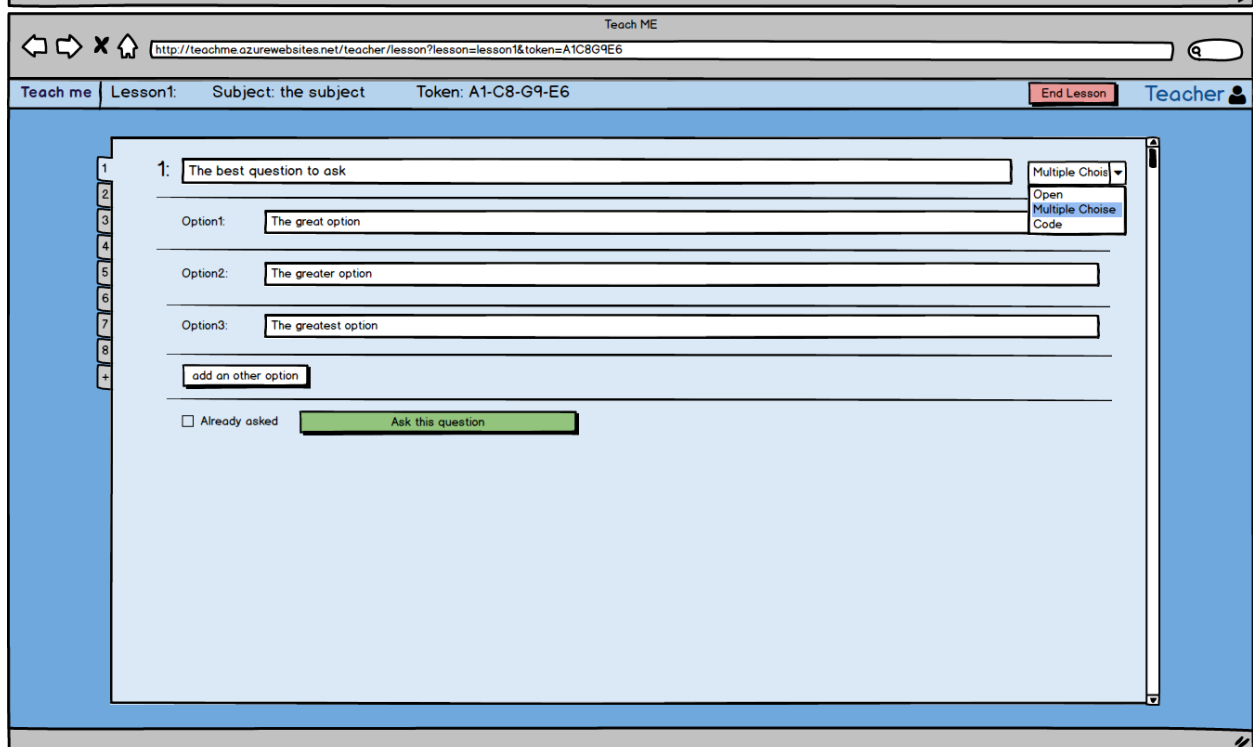
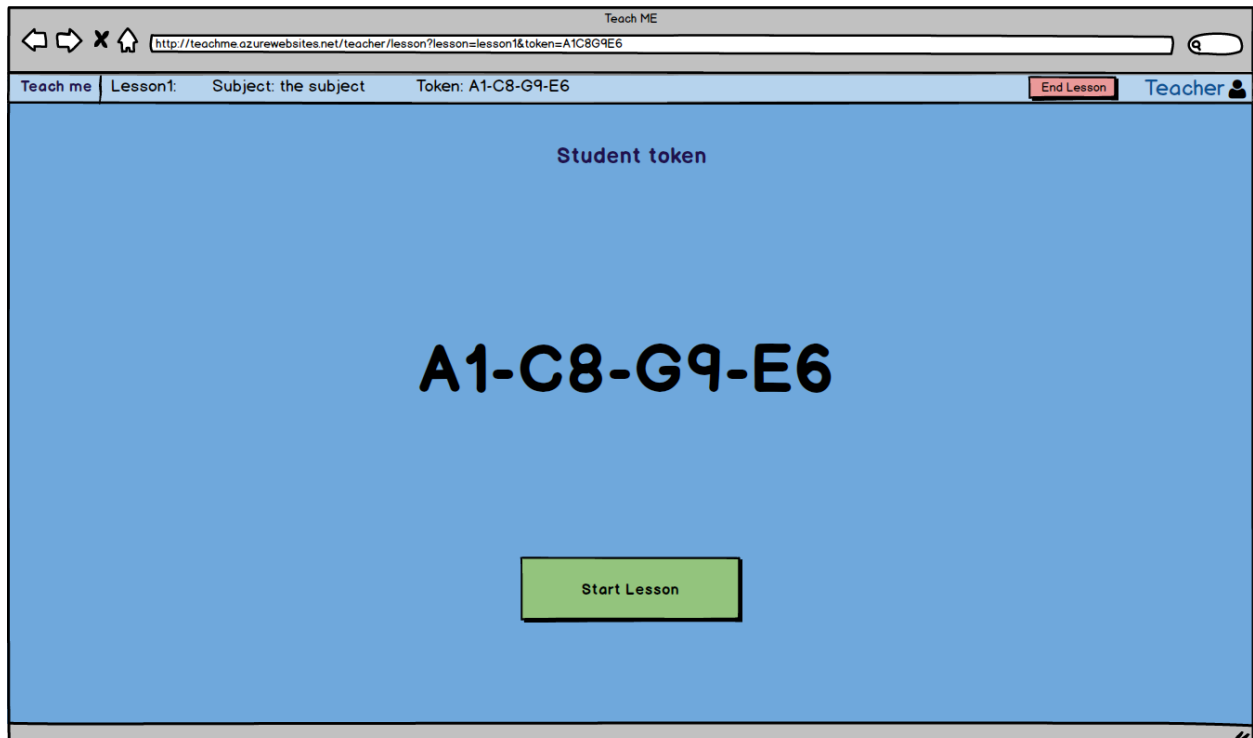
Code

question3

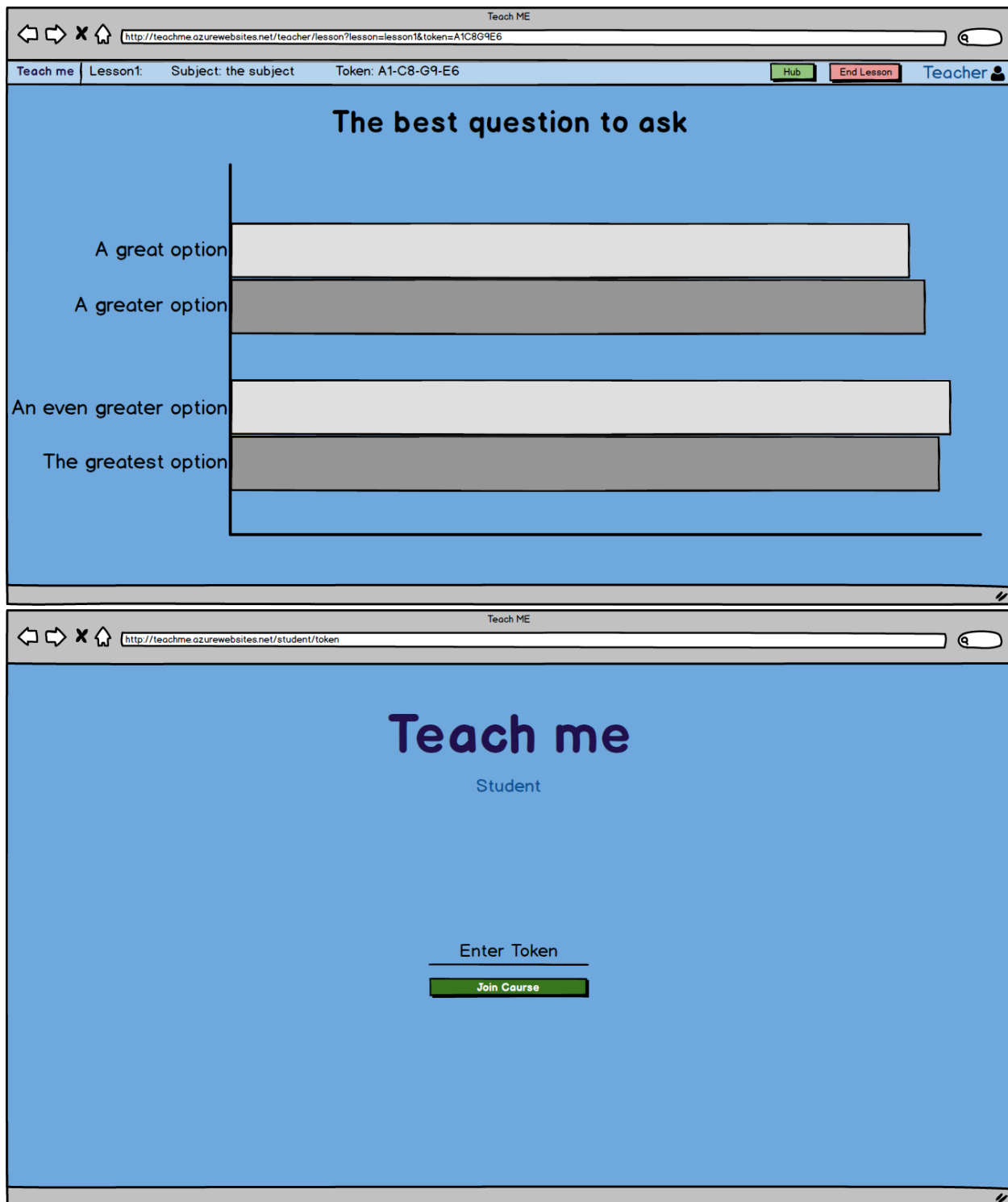
Code

Code example

add an other question







Teach ME

http://teachme.azurewebsites.net/student/form

# Teach me

Student

Course NAME  
Course SUBJECT

First name  
Last name

Only when the teacher  
deselects anonymous

Join Course

Teach ME

http://teachme.azurewebsites.net/teacher/lesson?lesson=lesson1&token=A1C8G9E6

Teach me | Lesson1: Subject: the subject | Student

Auto submit in: 10

## The best question to ask

- ◉ a great option
- a greater option
- an even greater option
- the greatest option

Submit

what's the difficulty level of this question?

★★★★☆

## Links

- *Github:*  
<https://github.com/Rival-Thompson/CloudApplications-TeachMe>
- *Github projects:*  
<https://github.com/Rival-Thompson/CloudApplications-TeachMe#boards?repos=68715541>
- *Mockups:*  
<https://balsamiq.com>
- *Studies Meteor:*  
<https://app.pluralsight.com/library/courses/meteorjs-fundamentals-single-page-apps/table-of-contents>  
<https://app.pluralsight.com/library/courses/meteorjs-web-application/table-of-contents>  
<https://www.meteor.com>
- *MongoDB*  
<https://mlab.com>