

Advanced JavaScript and Mobile Web

Academic Year – 2015 - 2016

SupTrac – Group Project



Advanced JavaScript and Mobile Web Project

Context

Your company needs a new ticket management application for its IT teams. This application has to manage two different roles: Product Owner & Developer.

Product Owners are the ones who have the business knowledge. They will use the application to create Tickets for each improvement or bug they'll want to submit to developers.

Developers will see the Tickets and work on them.

So your company needs you and your team to develop this application. Because it needs to be easy to develop and because you love it, you naturally chose to use Node.js with some HTML5 APIs.

This work must be done by a group of two or four students.

For this project, you must use the MEAN Stack: NodeJS, Express, MongoDB and AngularJS.

Specifications

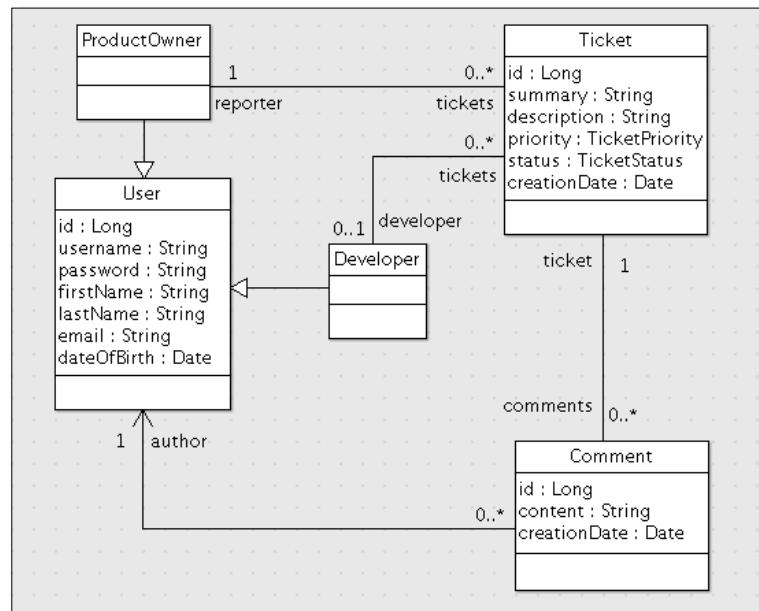
The first version of the website will be composed of several functionalities listed below:

- Authenticate an user
- Logout an user
- Display a page with the new tickets
- Display a page with all the tickets
- Display a ticket with its details and comments
- Post a new comment on a ticket
- For Product Owner users:
 - Create new tickets
 - Edit existing tickets
 - Delete existing tickets
- For Developers:
 - Change the state of a ticket to declare working on it
 - Change the state of a ticket to declare it done
 - Display all the tickets on which he declared work
- Send an email to the ticket reporter (Product Owner who created the ticket) when someone posts a comment on it. This option can be disabled by the ticket reporter.
- Send live notifications when a new comment or post is made



Advanced JavaScript and Mobile Web Project

1. Data Structure



The application will be composed of at least five entity classes:

- **User**, composed of at least **id**, **username**, **password**, **firstName**, **lastName**, and **email** fields
- **ProductOwner**, extending User and with a **tickets** field representing the list of reported tickets
- **Developer**, extending User and with a **tickets** field representing the list of tickets on which he work
- **Ticket**, composed of **id**, **summary**, **description**, **priority**, **status**, **creationDate**, **reporter**, **developer** and **comments** fields
- **Comment**, composed of **id**, **content**, **creationDate**, **ticket** and **author** fields
- You might want to add some entities depending on your needs.

Tickets can have three different statuses: **NEW**, **IN PROGRESS** and **DONE**.

Tickets can have five different priorities: **TRIVIAL**, **MINOR**, **MAJOR**, **CRITICAL** and **BLOCKER**.

Advanced JavaScript and Mobile Web Project

2. Authentication

The security is something very important in SupTrac. Each user of the platform must have an account to publish and view tickets. So, if anonymous user tries to display another page than the login form, he has to be redirected to it.

The screenshot shows a login form titled "Ticket Manager". The title is in a dark red header bar. Below it, the word "Login" is centered in a white area. There are two input fields: "Username:" followed by a text input box, and "Password:" followed by another text input box. Below these fields is a "Login" button. At the bottom of the page, there is a horizontal line and the text "Ticket Manager est un site de gestion de Ticket pour les projets IT."

For now, you only have to design the login page and **not the register one**. For your tests, you will insert users directly in Database.

When form is submitted, the application will verify that an account corresponding to the username exists and if it is, verify that the password is correct. If credentials are valid, store them in session and redirect user to the home page.

Note: As SupTrac cares a lot about security, you must store them as MD5 hash in your database.



Advanced JavaScript and Mobile Web Project

3. New Tickets

This is the page display just after the identification of the user. It lists all the tickets in the NEW state. The tickets have to be ordered by default from oldest to youngest. The user must be able to re-order the list by the field he wants.

When a new ticket is created or updated, this page must be updated automatically.

For each ticket, a link to display the detailed page has to be present.

Ticket Manager

New Tickets Add a Ticket All Tickets Logout

New Tickets

Ticket id	Summary	Priority	Reported by	Status	Action
1	Script to update Trac copyright date in license header of source files	MINOR	Henry Chef	NEW	Show
2	DOAP Integration	TRIVIAL	Henry Chef	NEW	Show
3	[Database schema] Ticket description field might be too short.	MAJOR	Henry Chef	NEW	Show
4	No version "0" for Wiki page "FooPage"	BLOCKER	Henry Chef	NEW	Show
5	[PATCH] Highlight New Events in Timeline	BLOCKER	Henry Chef	NEW	Show
6	support non-ascii filenames in mercurial vcs backend	MAJOR	Henry Chef	NEW	Show
7	[Patch] Display full user names instead of user login	MAJOR	Henry Chef	NEW	Show

Ticket Manager est un site de gestion de Ticket pour les projets IT.

4. All the Tickets

This page is identical to the previous one except it displays all the tickets, whatever the ticket's status.



Advanced JavaScript and Mobile Web Project

5. Display ticket details with comments

This page gives all the detailed information about a ticket.

Below, the list of all its comments ordered from the oldest to the youngest have to be displayed. Be careful, new comments must be received with (at your convenience) Server Sent Events or Web Sockets, in order to be directly viewable. When a new comment is displayed, a notification must be present in page title.

A button to add a new comment has to be present below this list.

Ticket #4

No version "0" for Wiki page "FooPage"		Opened:Nov 28, 2010 2:22:24 AM	
Reported by:	Henry Chef	Owned by:	Brice Argenson
Priority:	BLOCKER	Status:	IN_PROGRESS
Description: How to reproduce: Search for a non-existing page, i.e. "FooPage?" and click button "Create this page." From another browser window or another computer (same or different user) do the same, edit the page and click "Submit changes". Now, in the first window, whichever button you choose, you will get a Trac Error No version "0" for Wiki page "FooPage?".			
Stop Working			
From: Brice Argenson		Posted the: Nov 28, 2010	
This looks interesting, but calling env.get_known_users is costly. So for one this should only be done if the show_full_names option is selected. Then, caching that information in the Chrome instance is not a good idea since there's currently no way to get notified when this information changes. Recomputing the cache in each format_author makes the cache useless and is of course too costly. One interim solution might be to cache it in the req. A better solution should be found as part of the user API changes, planned for 0.13. I'm rescheduling this for 0.13, but feel free to propose an interim patch taking the ideas above (or better ones) into account.			
From: Brice Argenson		Posted the: Nov 28, 2010	
Sorry, not even in my head, I was referring to various ideas floating around (see #2456) and the fact that there's a general agreement that some improvement is needed in this area and that it should happen hopefully not later than 0.13...			
Add a comment			



Advanced JavaScript and Mobile Web Project

6. Add a comment to a ticket

Users must be able to add a comment on a ticket thanks to a simple form using AJAX.

Add a comment:

Ticket Manager est un site de gestion de Ticket pour les projets IT.

7. Create new Tickets

Product Owners only must be able to add new tickets to the application. When a ticket is added, his state is automatically set to NEW and the creation date set to the current date.
The link to go to this page has to be displayed only if the current user is a Product Owner!

Ticket Manager

[New Tickets](#) [Add a Ticket](#) [All Tickets](#) [Logout](#)

Add a Ticket

Summary:

Description:

Priority:

Ticket Manager est un site de gestion de Ticket pour les projets IT.



Advanced JavaScript and Mobile Web Project

8. Edit and Delete Tickets

When a Product Owner displays the detailed ticket page, he must have two links: one to edit the ticket and another to delete it.

This links have to be visible by Product Owners only! (but you must use the same HTML file)

Status:	IN_PROGRESS
Description:	
Edit Ticket Delete Ticket	
Posted the:Nov 28, 2010	

9. Report to work on a Ticket

When a Developer displays the detailed page of a ticket with his status to NEW, he must have a button for reporting to work on it. If he clicks on it, the ticket has to be associated with him and the status updated.

This button must be visible only on new tickets and only by developers! (but you must use the same HTML file)

When a developer displays the detailed page of a ticket associated with him and with his status to IN_PROGRESS, he must be able to report to finish working with another button.

From another mit changes". Now, ge "FooPage?".	Start Working
From another mit changes". Now, age "FooPage?".	Stop Working



Advanced JavaScript and Mobile Web Project

10. My Tickets

A developer must be able to display a list of all the in progress tickets on which he's currently working.

The screenshot shows a web application titled "Ticket Manager". At the top, there is a red header bar with the title "Ticket Manager" in white. Below the header, there is a navigation bar with four items: "New Tickets", "My Tickets" (which is highlighted in blue), "All Tickets", and "Logout". The main content area is titled "My Tickets" and contains a table with three rows of ticket information. The table has columns for "Ticket id", "Summary", "Priority", "Reported by", "Status", and "Action". The first row corresponds to ticket ID 3, the second to 4, and the third to 7. Each row includes a "Show" link under the "Action" column. At the bottom of the page, there is a footer note: "Ticket Manager est un site de gestion de Ticket pour les projets IT."

Ticket id	Summary	Priority	Reported by	Status	Action
3	[Database schema] Ticket description field might be too short.	MAJOR	Henry Chef	IN_PROGRESS	Show
4	No version "0" for Wiki page "FooPage"	BLOCKER	Henry Chef	IN_PROGRESS	Show
7	[Patch] Display full user names instead of user login	MAJOR	Henry Chef	IN_PROGRESS	Show

11. Send an e-mail when a new comment is post

We want to notify by e-mail the reporter of a ticket when someone adds a comment to it. The sending process must be asynchronous! Users should be able to disable this option.

12. Send notification when a new ticket or comment is post

We want SupTrac to refresh itself, so that the application stays as a tab in everyone browser. Use Server Events to display notifications on client side when a new ticket or comment is post.



Advanced JavaScript and Mobile Web Project

Instructions

- Plagiarism is forbidden.
- Make accessible his code on a public sharing platform (as GitHub) before the end of the evaluation is forbidden.

Don't abiding by these rules will result in suspension of your assessment and will be considered cheating.

Notation

Functionalities	Points
Authentication	2
New tickets listing	1
All ticket listing	1
Display a ticket with comments	2
Add a comment	2
Create a ticket	2
Edit & Delete a ticket	1
Report to work on a ticket	2
My tickets in progress listing	1
Send an asynchronous e-mail	2
Live notifications	2
User Experience	2
Code Quality	2
TOTAL	22

Appendix

For sending e-mail, you can use "smtp.free.fr" as SMTP server.

For code quality, just remember this quote:

"Always code as if the guy who ends up maintaining your code will be a violent psychopath who knows where you live" - Martin Golding



Advanced JavaScript and Mobile Web

Project

Return

You will return your graded exercise inside a ZIP archive named: **3WEB_SupTrac_IdBooster.zip**.

For example: 3WEB_SupTrac_10000.zip

Not respecting this convention will be sanctioned by penalties points.

Do not forget to add **your members list** in your ZIP archive.

Do not forget to add **your MongoDB file**.

You will send the archive **to your teacher SUPINFO email address** with 3WEB@supinfo.com in carbon copy field and **within the hours assigned to do the project**. After that delay, your project **will not be corrected and the mark 0 will be assigned to you**.

The project must be handed in before 14 days at 23:59 (Paris time) after reception.

