



QA Cinemas

Team Mystic



Introduction and roles

Daniel - Scrum master and front-end developer

Gabriel - Front-end developer

Lukas - Product owner and back-end developer

Sam - Back-end developer

Solomon - Front-end developer



Meeting MVP (1 of 2)

Went through the spec systematically as a team:

Home page

The QA Cinemas site needs a home page. The home page shall:

- Be generally attractive
- The page shall be the default for the entire site
- Starting from the home page, site users may navigate to other areas of the site
- The site should have a picture or graphic evocative of the movies or the cinema on it
- Communicate to the viewer the purpose of the site

Starting by visualizing how we wanted the page to look

Agreeing on a consistent theme which would be used throughout the website.



Meeting MVP (2 of 2)

We used MoSCoW prioritisation as a team.

I.e. to get an early sense of what would be nice to have/ could have in addition to should have.

Screens

The site has a page dedicated to the screens

- Include image of the seating plan of the standard screen
- Include an image showing the décor for the standard screen
- Include image of the seating plan of the deluxe “Directors Box” screen
- Include an image showing the décor for the deluxe screen

Ticket Bookings

The site should include a page dedicated to booking tickets.

The page is part of the overall site navigation

- Booking should include the following information:
- Movie title
- Screening date and time
- Name of booker
- Number of seats
- Adult
- Child
- Concession



Allocating roles

- Everyone talked about what their preferences were based on the content learned so far
- Went through everyone's strengths/what we were most comfortable doing
- Helped give everyone idea what roles we will be doing for the project
- This also meant we can all work efficiently by sticking to what we were comfortable with

Jira Board

Type	Key	Summary	Assignee	Reporter	P	Status	Resolution
	QSPC-27	Home Page Must Exist	Daniel Li	Lukas Rygh	=	DO NOT USE	Done
	QSPC-25	About Page	Unassigned	Lukas Rygh	=	DO NOT USE	Done
	QSPC-24	Site Search					
	QSPC-22	Site Navigation					
	QSPC-21	Comment moderation					
	QSPC-20	Movie Topic Discussion					
	QSPC-17	Getting there					
	QSPC-16	Local Venues					
	QSPC-15	Paying for a booked ticket					
	QSPC-13	Ticket Booking					
	QSPC-12	Film Screen Descriptions					
	QSPC-11	Film Classifications					
	QSPC-10	New Release Gallery OnClick					

Minimum Viable Produ... / QSPC-10

New Release Gallery OnClick

Attach

Add a child issue

Link issue



Description

As a User

I want to be able to click on the new release posters and find out when they come to the cinema

So I know when to come and watch them and who is in it.

	QSPC-9	New Releases Gallery					
	QSPC-8	Cinema opening times					
	QSPC-7	Listing Gallery Poster OnClick	Daniel Li	Lukas Rygh	=	FINISHED	Unresolve
	QSPC-6	Listing Gallery Posters	Unassigned	Lukas Rygh	=	DO NOT USE	Done
	QSPC-5	Home Page navigation	Gabriel	Lukas Rygh	=	FINISHED	Unresolve
	QSPC-4	The Background Cannot be White	Solomon Boundy	Lukas Rygh	=	DO NOT USE	Done
	QSPC-2	Email Form for Contact Page	Unassigned	Lukas Rygh	=	DO NOT USE	Done
	QSPC-1	Email Contact for site owner	Gabriel	Lukas Rygh	=	FINISHED	Unresolve

Projects / QA Specialism Project - QA Cinemas

QSPC Sprint 2

1 day remaining Complete sprint

QSPC-70

Optional Front-End testing (Selenium and Jest)

QSPC-68

all API Keys and Passwords must be stored in a .env

Presentation

QSPC-69

Back-end testing to have 80% coverage.

README MUST EXIST

MINIMUM VIABLE PRODUCT

QSPC-55

mail Contact for site owner

MINIMUM VIABLE PRODUCT

QSPC-1

sting Gallery Pages

MINIMUM VIABLE PRODUCT

QSPC-28

sting Gallery Poster OnClick

MINIMUM VIABLE PRODUCT

QSPC-7

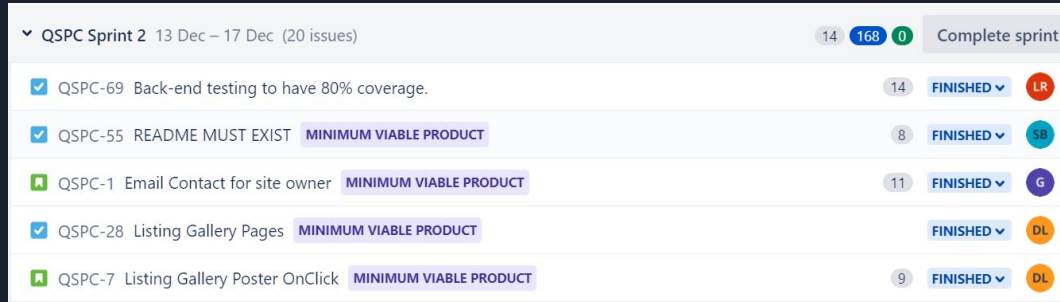
Assignee	Reporter	P	Status	Resolution
Samuel Sheldon	Lukas Rygh	=	DO NOT USE	Done
Unassigned	Lukas Rygh	=	DO NOT USE	Done
Gabriel	Lukas Rygh	=	DO NOT USE	Done
Gabriel	Lukas Rygh	=	DO NOT USE	Done
Unassigned	Lukas Rygh	=	DO NOT USE	Done
Solomon Boundy	Lukas Rygh	=	DO NOT USE	Done
Daniel Li	Lukas Rygh	=	DO NOT USE	Done
Daniel Li	Lukas Rygh	=	DO NOT USE	Done
Gabriel	Lukas Rygh	=	DO NOT USE	Done
Solomon Boundy	Lukas Rygh	=	DO NOT USE	Done
Daniel Li	Lukas Rygh	=	DO NOT USE	Done
Gabriel	Lukas Rygh	=	DO NOT USE	Done
Daniel Li	Lukas Rygh	=	FINISHED	Unresolve

Allocating tasks using Jira board

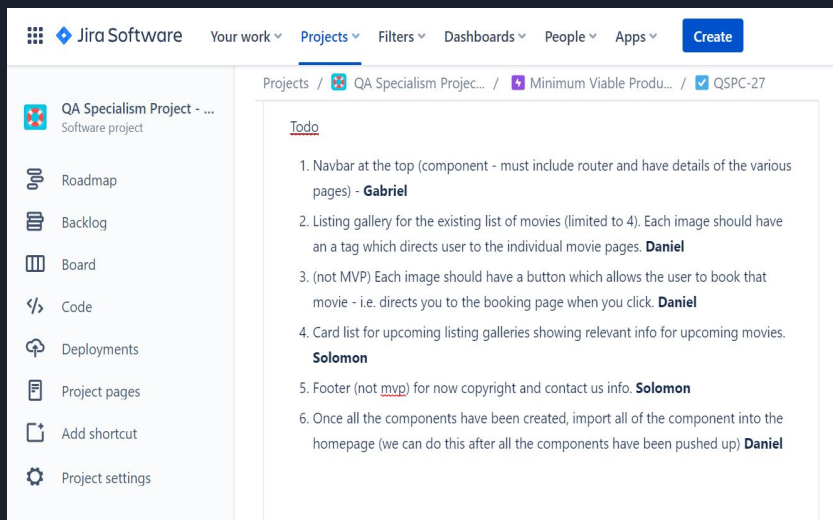
After we allocated roles and created user stories and basic tasks on the jira board, we then split into front and back end teams and wrote down detailed descriptions associated with each task coming to an agreement on who would do what and how we best get there.

By using this method, each member of the team had a clear roadmap on what to do in order to achieve MVP.

After our first week retrospective we spent each morning going over the tasks on the Jira board and assigning them where appropriate, to better track our progress and ensure we were working at a good pace.



Homepage



The screenshot shows the Jira Software interface. The top navigation bar includes 'Jira Software', 'Your work', 'Projects', 'Filters', 'Dashboards', 'People', 'Apps', and a 'Create' button. The left sidebar lists navigation options: 'QA Specialism Project - ...', 'Roadmap', 'Backlog', 'Board', 'Code', 'Deployments', 'Project pages', 'Add shortcut', and 'Project settings'. The main content area displays the project breadcrumb 'Projects / QA Specialism Project / Minimum Viable Product / QSPC-27' and a 'Todo' section with a list of six tasks.















QA Specialism Project - ...
Software project

Projects / QA Specialism Project / Minimum Viable Product / QSPC-27

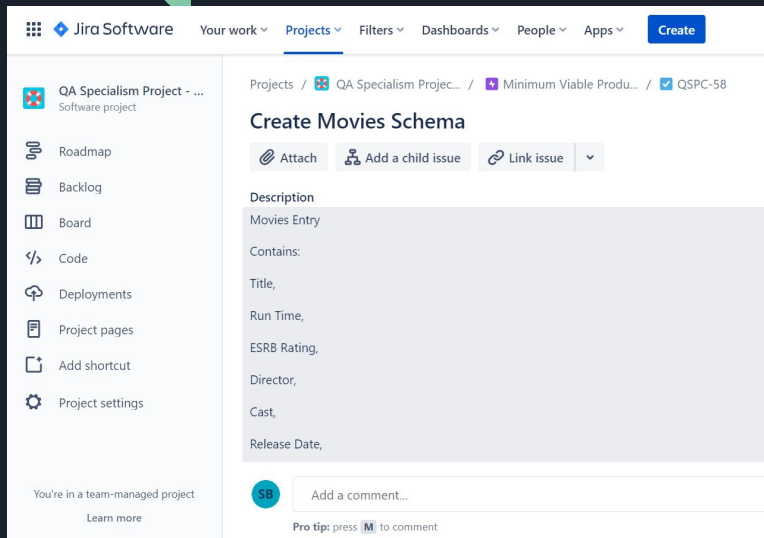
Todo

1. Navbar at the top (component - must include router and have details of the various pages) - **Gabriel**
2. Listing gallery for the existing list of movies (limited to 4). Each image should have an a tag which directs user to the individual movie pages. **Daniel**
3. (not MVP) Each image should have a button which allows the user to book that movie - i.e. directs you to the booking page when you click. **Daniel**
4. Card list for upcoming listing galleries showing relevant info for upcoming movies. **Solomon**
5. Footer (not mvp) for now copyright and contact us info. **Solomon**
6. Once all the components have been created, import all of the component into the homepage (we can do this after all the components have been pushed up) **Daniel**

Tasks

Key	Summary	Assignee	Reporter
QSPC-38	About Page Must Exist	 Gabriel	 Lukas Rygh
QSPC-36	Discussion Board Must Exist	 Unassigned	 Lukas Rygh
QSPC-35	<u>"How to get here" Page must Exist</u>	 Solomon Boundy	 Lukas Rygh
QSPC-34	Payment Page Must Exist	 Daniel Li	 Lukas Rygh
QSPC-33	Booking Page Must Exist	 Daniel Li	 Lukas Rygh
QSPC-32	Cinema Screen Listing Page Must Exist	 Gabriel	 Lukas Rygh
QSPC-31	Film Classification Page Must Exist	 Solomon Boundy	 Lukas Rygh

Schema



The screenshot shows the Jira Software interface for a project named 'QA Specialism Project - ...'. The left sidebar contains navigation links: Roadmap, Backlog, Board, Code, Deployments, Project pages, Add shortcut, and Project settings. The main content area displays the 'Create Movies Schema' issue page. At the top, there are tabs for 'QA Specialism Project...', 'Minimum Viable Produ...', and 'QSPC-58'. Below the tabs, there are buttons for 'Attach', 'Add a child issue', and 'Link issue'. The 'Description' section shows the issue title 'Movies Entry' and a list of fields: 'Contains:', 'Title,', 'Run Time,', 'ESRB Rating,', 'Director,', 'Cast,', and 'Release Date,'. At the bottom, there is a comment box with a '5B' icon and the text 'Add a comment...'. A 'Pro tip' message at the bottom right says 'press M to comment'.

Projects / QA Specialism Project... / Minimum Viable Produ... / QSPC-58

Create Movies Schema

Attach Add a child issue Link issue

Description

Movies Entry

Contains:

Title,

Run Time,

ESRB Rating,

Director,

Cast,

Release Date,

5B Add a comment...

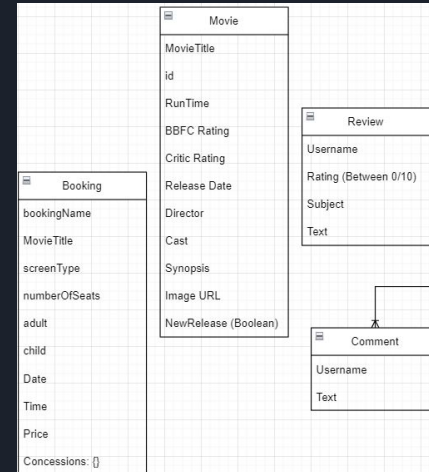
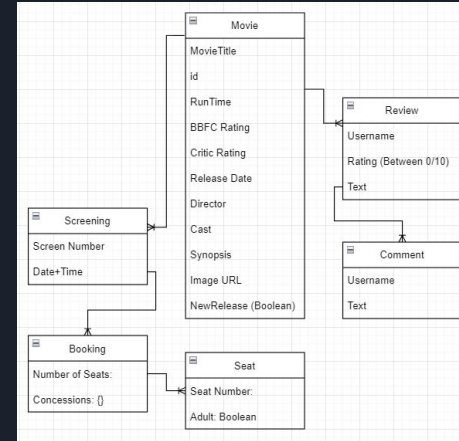
Pro tip: press M to comment

Tasks

Key	Summary	Assignee	Reporter
QSPC-67	Use ENV variables for information access	LR Lukas Rygh	SS Samuel Sheldon
QSPC-66	Setup Atlas For MongoDB	SS Samuel Sheldon	SS Samuel Sheldon
QSPC-62	Create Comment Schema	LR Lukas Rygh	SS Samuel Sheldon
QSPC-61	Create Review Schema	SS Samuel Sheldon	SS Samuel Sheldon
QSPC-59	Create Booking Schema	SS Samuel Sheldon	SS Samuel Sheldon
QSPC-58	Create Movies Schema	SS Samuel Sheldon	SS Samuel Sheldon
QSPC-50	Stripe API used on Payment Page	SS Samuel Sheldon	LR Lukas Rygh











Back-End Schema

- Initial Database built from a relational perspective
- MongoDB isn't relational in the same way, and linking was unnecessary for a lot of our proposed tables
- Eventually cut down to the second version, which fulfills all requirements and achieves MVP.



Github and Feature-Branch Model

- All code was hosted on the same repository, using the feature-branch model to avoid conflicts.
- All pull requests went through lukas who had to approve them to ensure no merge issues and limit the risk of broken code being uploaded.
- We also used smart commits to track progress on Jira alongside our progress with the code itself.

Active branches			
dev	Updated 7 minutes ago by Lukas Rygh		 
navbar_tweaks	Updated 1 hour ago by Man Chung Daniel Li	#47	Merged  
formattingFixes	Updated 17 hours ago by gabejose	#46	Merged  
searchfeature	Updated 18 hours ago by Man Chung Daniel Li	#45	Merged  
contactPageFix	Updated 19 hours ago by Lukas Rygh	#44	Merged  
View more active branches >			



Key security and .env files

- .env files normally store data that cannot or should not be uploaded onto repositories or exposed to the user.
- React .envs are inherently insecure and accessible through the Inspect Element function - unless hosted on a backend that the main site simply sends requests to.
- Outside of security, .env files also provide an easy way for later owners to easily adapt functionality to their own database or systems.