Project Exam 1 Rapport

Introduction/Interpretation of the assignment

My interpretation of the assignment was to create a themed microsite that will focus on one part of SpaceX's work in order to promote their space programs. For this project the main goal is to create a microsite that focuses on SpaceX's mission to mars program. The goal of the site is to collect all information regarding the mission to mars under one comprehensive domain to better present the program to the users. I wanted to create a microsite that was informative and also visual so that the text wouldn't be overwhelming for the user. I wanted the site to encourage further reading as well as keeping up to date on the program. Since microsites are often build around a certain event, I wanted created a call-to-action to guide the users towards the newsletter that will keep them updated on the project.

Project Plan

Part 1 planning:

Milestone Title	**Description**	**Date**
Functional Specifications	Figure out the functional specifications for the site.	13/05/19 - 14/05/19
Mock-up	Create a mock-up to give an idea of how the website will look.	15/05/19 - 15/05/19
Gantt Chart	Create Gantt chart for overview of the project timeline.	16/05/19 - 16/05/19
research SpaceX	Research SpaceX and mission to mars to help in creating of the layout and content of the website.	13/05/19 - 17/05/19
project planning document	Create project planning document to give insight into how the project will be executed.	16/05/19 - 16/05/19

Part 2 Design:

Milestone Title	**Description**	**Date**
Target Audience	Market research and find out who the target audience is.	20/05/19 - 20/05/19
Create Personas/Storyboards	Create personas/storyboards to better understand the users.	20/05/19 - 20/05/19

Choose API	Choose which API is best most appropriate for the site.	22/05/19 - 22/05/19
Create Prototype/wireframe	Create a prototype of the whole site.	22/05/19 - 22/05/19
Create Site Structure	Create the layout and site structure.	23/05/19 - 24/05/19

Part 3 Build:

Milestone Title	**Description**	**Date**
Html and CSS	Start building the foundations of the site.	27/05/19 - 04/06/19
Implementing JavaScript	Add JavaScript to the site.	05/06/19 - 07/06/19
UX/UI Design	Create user experiences on the site.	06/06/19 - 07/06/19
Add Text and Photos to Website	Add text and edit photos to the design.	07/06/19 - 07/06/19
Test/adjust Responsiveness	Make sure the site is responsive on all platforms.	10/06/19 - 11/06/19

Part 4 Launch:

Milestone Title	**Description**	**Date**
Cross Platform and Browser Testing	Test that the site is compatible with all platforms	12/06/19 - 14/06/19
UX/UI Testing	Test the UX design.	13/06/19 - 13/06/19
Refinement	Make final refinements	14/06/19 - 14/06/19
Bug Fixes	Fix bugs that may have occurred.	14/06/19 - 17/06/19
Launch Site	Launch the site.	17/06/19 - 17/06/19

Theme

Build a microsite about the space program mission to mars. A one stop site for all information related to the project.

Features

- All information on one site.
- Display informative home screen.
- Display timeline of project components.
- Contact form for user questions and contact.

- Newsletter sign up for updated information.
- Menu for navigation.

Epics

As a user, I want to access all information related to the mission on mars program on one site.

As a user, I want to get the latest news about the program so I can keep up to date.

As a user, I want to know what time there will be a launch and when I can expect progress to be made.

As a user, I want to be able to navigate the site to find the information I need.

As a user, I want to be able to contact SpaceX about their project.

User stories

User story 1:

User is a student doing a school project about the mission to mars. Therefore the user is curious about the scope of the mission and needs a comprehensive timeline of the project.

User story 2:

User is passionate about space exploration and wants to keep up to date on everything about the mission. Therefore the user wants to be able to sign up for a newsletter so they don't have to log on to the site regularly to see if there has been any updates.

User story 3:

User is very interested in the technical design of the Mars transportation infrastructure, and wants to talk to someone that works for SpaceX.

Technical Specification

Overview

The goal for this site is to make it a one stop site for the user to get information about SpaceX mission to mars. The site should be responsive and have a flow that inspires further reading.

List of Technical Specifications

- Given the small scope of the site and considering I am the only one working on the site, CMS is not necessary at this time.
- Site should be responsive and convey all the information no matter the device used to view the site
- The site should load quickly, and not be unnecessarily large.

- The site should get all the necessary info about the company, mission and rockets from the SpaceX API.
- The forms should validate all input, before sending the data.

1. Visitor Interaction

Req	Description	Comments
1.1	As a user I want an easy to use menu, to navigate the site	Create a simple and easy to use menu
1.2	As a user I want a fun and easy to navigate timeline of the mars mission	An interactive timeline of the Mars mission
1.3	As a user I want a way to contact SpaceX with questions about the mars mission	Create a contact form
1.4	As a user I want a way to stay up to date on the latest Mars mission news	Create a form that allow users to subscribe to a news letter
1.5	As a user I want to be able to store my user information for when I may have further questions	Store information sent from contact form, for later use
1.6	As a user I want to see the latest mission info on test launches for the rockets intended to be used for the Mars mission	Get the latest launch data from the API
1.7	As a user I want to see relevant company info	Get company info from the API

2. Sitemap and Navigation

Req	Description	Comments
3.1	Each page will have navigational links in the header menu	
3.2	Each page will have navigation links in the footer	
3.3		

3. Tracking (website statistics)

Req	Description	Comments
5.1	https://analytics.google.com	

4. Search Engine Optimisation (SEO)

Req	Description	Comments
6.1	Headlines, meta tags, and navigational elements	
6.2	Google ad words	
6.3	Clear text and theme, use tag words in text.	
6.4	Semantic HTML5	
6.5	Small file sizes, and good performance to increase google score	

5. Accessibility

Req	Description	Comments
8.1	Make sure the color contrast follows WCAG specifications	
8.2	Use a readable font	
8.3	Accommodate for keyboard navigation	
8.4	Make website responsive, so that it can be accessed from any device	

FUNCTIONAL SPECIFICATION DOCUMENT

<SPACEX MICROSITE>

<16.06.2019>

Table of Contents

1. Introduction

Build a self-contained site for SpaceX about the missions to mars program activity on its own domain. The purpose of this microsite is to generate buzz and interest around the SpaceX Mission, by showing the status and latest news for the mission.

1.1 Purpose of the document

The purpose of this document is to provide detailed information on how the system solution will function and the requested behaviour. This document is created based on the high-level requirements identified in the Business Requirements Document and provides traceability on the functional specifications back to the business requirements. Included in this document will be the detailed functional requirements including use cases, system inputs and outputs, process flows, diagrams, and mock ups.

1.2 Project Scope

Create a microsite for SpaceX to raise awareness about space program activity around the world. The site should appeal to the specific target audience and provide links to more information. The site should be responsive, and function well on a variety of platforms. The site should include a timeline of information about space program activity. The site should have a contact form for the users. The site should be easy to use. And conform to WCAG standards. The site should give the user an option to sign up for the newsletter to stay update on the Mars mission.

1.3 Risks and Assumptions

Risk	Level	Impact	Responsible
Site will not be responsive on all platforms and browsers.	High	Users will not be able to see all content on their devices.	Development Team
Links provided will not lead to the intended destination.	Medium	Links will not lead the users to the correct updated information.	Development Team
JSON API dynamic data is not employed as intended.	High	The date doesn't show up on the site the way it is supposed to.	Development Team
Contact form does not validate information	Medium	The user is not alerted that the information they added is incorrect and the company is sent the wrong information and cannot contact the user with a response.	Development Team
Color contrast on text does not conform to WCAG standards	High	The users wont able to read text and other information, due to low contrast ratios	Design Team

2. Functional Specifications

The system will need its own domain. The page should be responsive on all platforms, and provide links to live feeds and information. The site should have a contact form with JavaScript validation. A timeline/calendar for the mission should be available. The site needs to be easy to use and well-designed. The site needs to be informative and cover all the information related to the mission to mars. The site should be a one stop site for all the information regarding this mission for SpaceX. The design needs to be appealing and the information needs to be displayed in an elegant way.

Specifications	Comments
Create new domain	
Add contact form	Add Regex validation to form inputs
Add JavaScript validation	To make sure the information added is correct
Employ JavaScript/JSON API for dynamic data and construction and styling of HTML/CSS	To add dynamic data and construction to the site.
Add timeline/calendar of relevant events	To give the user an overview of the program.
Git hub repository needs to be created	To add all relevant code for this project.
Newsletter subscription form	Add a newsletter subscription form so the user can get the latest news on email

2.1 <Develop microsite about the mission to mars program>

2.1.1 Purpose/ Description

The goal is to create a microsite about the mission to mars program and its progress. The site should include all relevant information about the program as well as provide a way for the users to contact SpaceX and interact with them.

2.1.2 Use case

UC-1 wants specific information about mission to mars.

A user has heard about SpaceX's mission to mars through the media and wants to learn more. The user is interested in mars and the exploration of mars and wants to know more about SpaceX's work on the project.

UC-2 heard about SpaceX, wants to know what they are working on.

A user is on the SpaceX main site and wants to learn what they are working on, accesses the microsite through the main site to learn more.

UC-3 possible investor in SpaceX wants to learn more.

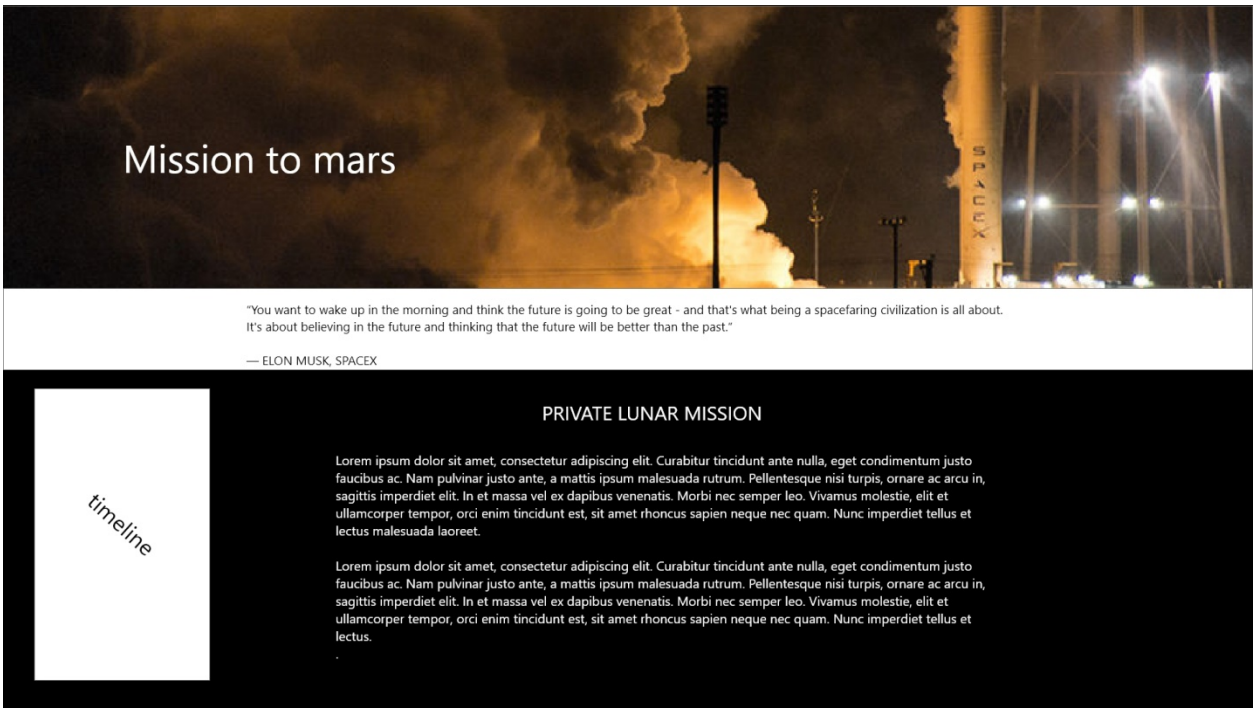
A user is considering buying stock in SpaceX, wants to know more about what they are about as a company and what they are working on.

UC-1	**UC-1 wants specific information about SpaceX mission to mars.**
Primary Actor(s)	Readers
Stakeholders and Interest	CEO Business Owner SpaceX
Trigger	Create microsite

Pre-conditions	Website does not exist
Post-conditions	URL of website to exist
Main Success Scenario	1. search for information about mission to mars 2. find microsite 3. Read relevant Information 4. User stays on the page and interacts with several of the components.
Extensions	If URL does not work, try google search for mission to mars spaceX
Priority	High
Special Requirements	A domain name and host to exist
Open Questions	

3. Mock-up

Img 1.1 mock-up of FrontPage on desktop.



4. Integration Requirements

The only integration for this microsite will be the SpaceX API.

4.1. Exception Handling/ Error Reporting

The microsite will log, or warn the user if there is something wrong with the fetching of the data from the API. While the API will provide error codes and error statuses.

Target audience/research

a. Target audience

For this assignment I did research on SpaceX to try and figure out who their target audience would be. Considering the language used on their site witch is technical in nature, but also simplified in a way that most people could understand it, I would say their main target audience is technology and space interested adults, but also trying to reach as many people as possible. SpaceX is a privately owned company with great ambitions. SpaceX has gained worldwide attention for a series of historic milestones. It is the only private company capable of returning a spacecraft from low Earth orbit, which it first accomplished in 2010 (https://www.spacex.com/about). Their ultimate goal is to enable people to live on other planets. With this said their product isn't at this time available to everyone as it comes with a massive price tag, falcon 9 costs 62 million dollars to launch, that most people cannot afford or have a use for. This part of their company I would say is mostly directed at the extremely wealthy, large companies and the government. That being said I would say they consider most people within their target audience, because their main goal is to make life multiplanetary and they are playing the long game. Because of this I would say the main target audience for the microsite would be educated, middle to upper class men and women who can see share in their vision, but also wanting to reach out to anyone and everyone who is interested in the mission to mars. Therefore the site should be informative, but also make it easy for most people to understand the mission. The main focus of the microsite is to promote SpaceX's mission to mars to the public, but also to represent SpaceX as a whole for potential investors.

1. **Personas and Storyboards**
    a. **Personas**

Based on my research of SpaceX, I used what I learned to create 3 personas that could represent typical user's login on to the microsite.

Hannah

- Age 29
- Live-in-partner
- Works at a university
- University graduate

Goals: For Hannah the web is a great place to conduct research and get information. She wants to be able to get a comprehensible understanding about the project without having to leave the site for more information.

Usability needs: straightforwardly and informative. Needs to have a clear layout to where she can find information she is looking for.

Bookmarks: science illustrated, space.com, various news outlets.

Luke

- Age 35
- Married
- Works in an office
- College graduate

Goals: Wants to find information about life on mars. Luke is a practical person and wants to know how and when life on mars could be possible. He isn't to interested in the technical aspects of the spaceship as much as he is fascinated by SpaceX's ambitious plan.

Usability Needs: Clear timeline of mission and general scope of mission to understand the timeline.

Bookmarks: travel sites, home improvement sites and wall street journal.

Harrison

- Age 45
- Married with children
- Management at a large television corporation

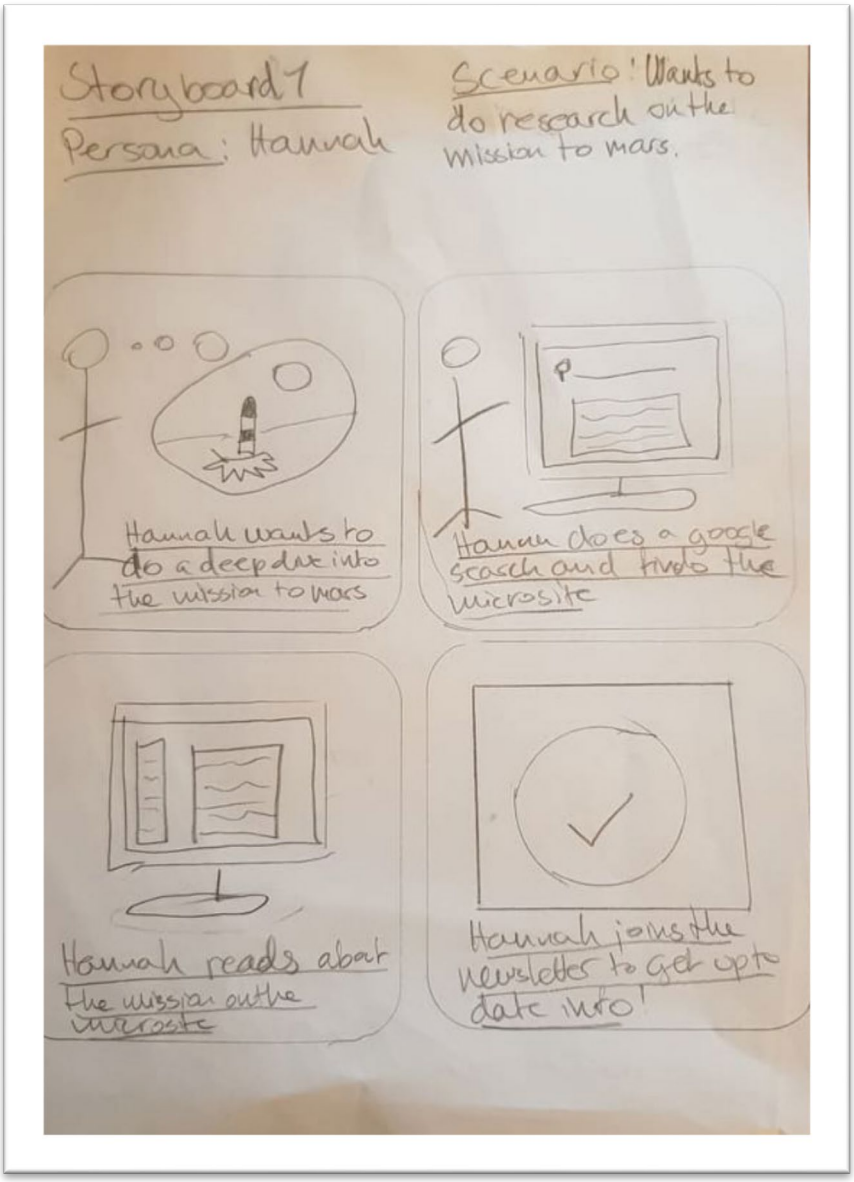
Goals: to learn about SpaceX as a company and its mission from curious, but business viewpoint.

Usability Needs: simple user interface that makes it easy to find the categories that he wants. He is not particularly tech savvy and need to keep it simple otherwise he loses interest.

Bookmarks: various news sites and shops.

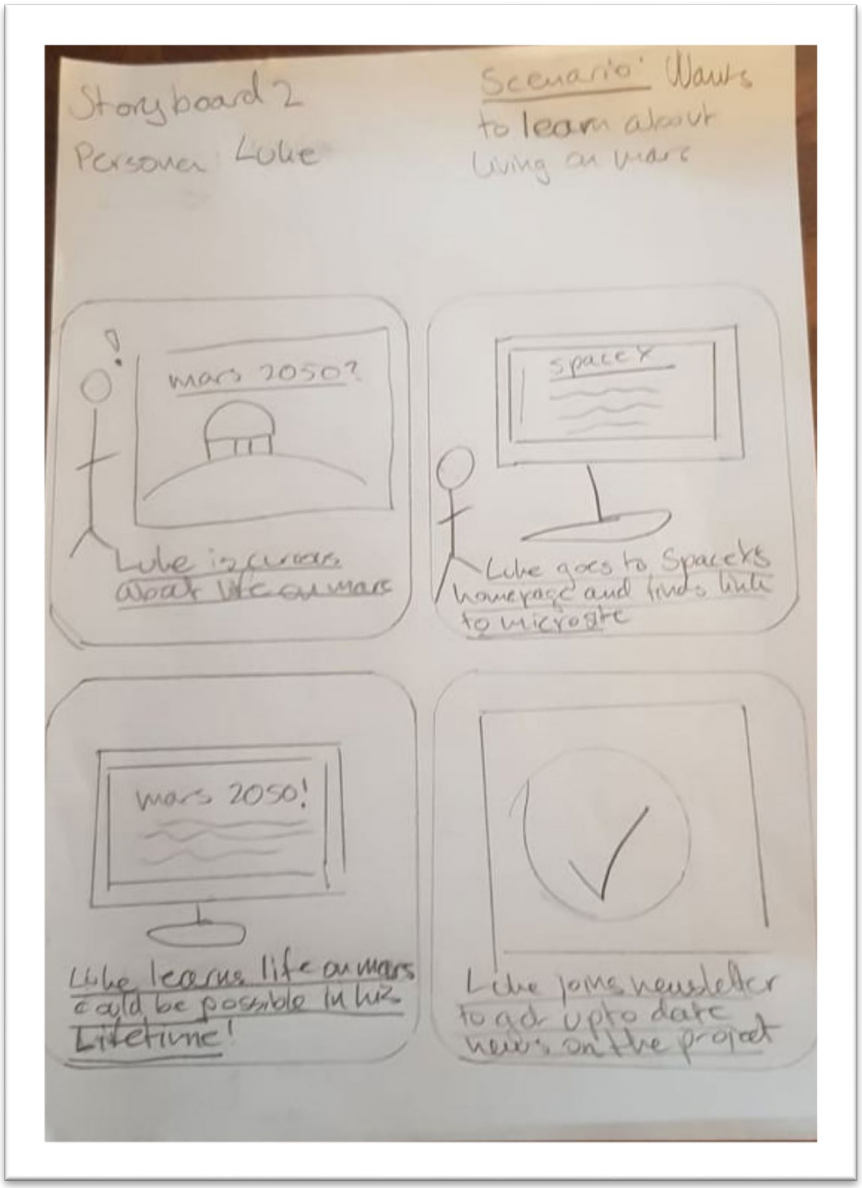
b. Storyboards

Storyboard 1



Hannah has heard about the possibility of humans living in space before, but doesn't know anything about SpaceX's project. Hannah does a google search on the subject and wants to do more research into it. Hannah clicks the link to the microsite and reads about the mission. After this Hannah wants to keep up to date with the progress of the mission and joins the newsletter.

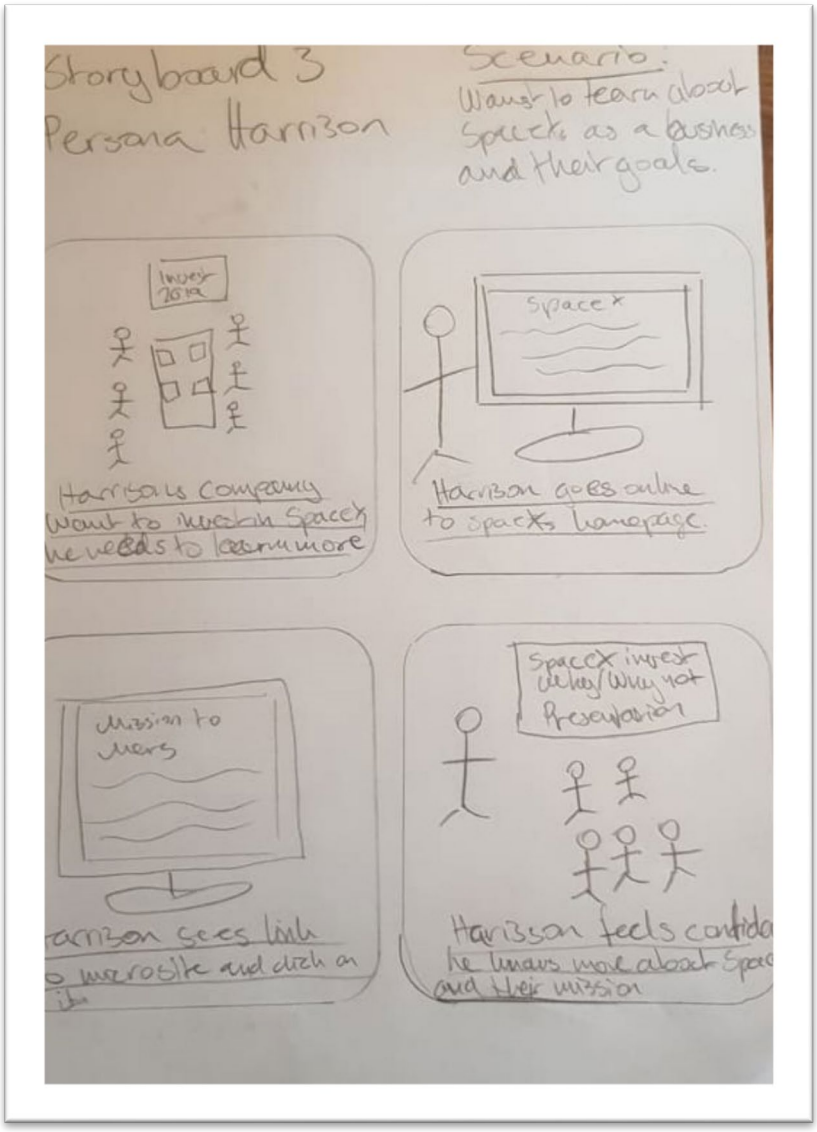
Storyboard 2



,

Luke is curious about humans living on mars and wants to know if it is possible in his lifetime. Luke knows about SpaceX, but not their mission to mars so he goes on their website to see what I can find. He finds a link to the microsite and clicks it to learn more. Luke joins the newsletter to get up to date information on the project.

Storyboard 3



Harrisons Company wants to invest in SpaceX so he needs to do more research in the company. He goes to SpaceX's homepage; there he sees a link to the microsite and clicks on it. The site gives Harrison an insight into SpaceX's ongoing project and also how they market it themselves. Harrison feels confident in the information he has gathered and feels he knows what SpaceX is about.

b. Wireframe/ prototype

Link to prototype: https://xd.adobe.com/view/30bdc821-38e3-472f-79de-695c40e6a798-468c/

When creating the design I tried to keep in mind that it is a microsite and most microsites are more visual, than for example a homepage for a company. So I wanted the images to play a bigger role when creating the design, and wanted to create a website that with a good flow as to give the illusion that all the pages are part of each other and the theme. Also when doing research I found that most microsites are very focused on images and flow which helps bring the focus to the theme of the site and the visit to the page is more about the journey than looking for specific information like you would on a product homepage. I noticed microsites don't have as much information above the fold as you would find on a different kind of site. The microsites encourage scrolling and have minimal text above the fold. Microsites remove the distractions that integration into the main corporate site would bring. Primary site navigation is removed, as well as footers and other irrelevant elements (https://boagworld.com/marketing/microsite/).

Interface Design: Affordances, Navigation, Persuasion

When creating the layout I took the information I learned from my research and tried to think of the most logical use for the website and the thought process one might have when entering the site. I decided that the first thing the user should see is the call-to-action because that is one of the main reasons for guiding the user to the microsite. Also I wanted the frontpage to have some elements from the other pages that could guide the user to the other pages and read about the parts of the mission they are most interested in. In that way the frontpage also functions as a summary of the microsite, with prompts to lead you to further reading. The join our newsletter button is also featured on the other sites. Since microsites try to keep distractions to a minimum I decided to only have a just the top navigation bar for the microsite and remove the footer to minimize distractions (https://boagworld.com/marketing/microsite/).

I decided to keep the hero image the same on all pages to help create a better flow on the website and give it a better consistency.

Graphic design

When creating the design I tried to keep in mind that it is a microsite and most microsites are more visual, than for example a homepage for a company. So I wanted the images to play a bigger role when creating the design, and wanted to create a website that with a good flow as to give the illusion that all the pages are part of each other and the theme. Also when doing research I found that most microsites are very focused on images and flow which helps bring the focus to the theme of the site and the visit to the page is more about the journey than looking for specific information like you would on a product homepage. I noticed microsites don't have as much information above the fold as you would find on a different kind of site. The microsites encourage scrolling and have minimal text above the fold. Microsites remove

the distractions that integration into the main corporate site would bring. Primary site navigation is removed, as well as footers and other irrelevant elements (https://boagworld.com/marketing/microsite/).

One of the most significant problems with microsites is that they force users to adapt to different user interfaces (https://boagworld.com/marketing/microsite/). I wanted to try and avoid this by keeping interface as simple and easy to navigate as possible while also keeping it within the spirit of a microsite. I also kept in mind the layout of SpaceX's homepage when deciding on the design since many user could theoretically be navigating from that site if this was an actual microsite made by SpaceX. I also feel the black design gives a website a sleek and intriguing look that is suitable for this topic. That being said, a completely black website is hard to pull off, black is the heaviest colour after all and

Colour

When choosing the colours I wanted to keep with the dark theme that SpaceX already has, but also make it slightly different to make it clear that it is a different site with a different purpose. I also chose to keep with the a black design because it is fitting to the space and exploration theme. Black website designs can be elegant and mysterious which I find fits the theme well. The biggest user-experience problem with a dark background is readability, not just because it is harder for our eyes to notice the colour difference, but also because of the screen light that makes the white shine (https://dev.to/neshaz/how-and-when-to-create-a-website-with-a-black-background-design-2a3e). To avoid this and make the website more pleasing to the eye I decided to go with more grey tones in my palette to make the difference less dramatic and also giving me more options for the different elements on the site, so that I had more options for making them stand out to the user. To highlight my calls-to-action I decided to add an orange/yellow button to make the buttons stand out, I also added this colour on different elements to make them pop on the page and also make the colour a bigger part of the overall design and give the page a better flow.

Logo

When creating the logo I wanted it to fit with the colour palette of the site and also the theme. I decided to go with a rocket design to fit with the spacecraft launch theme. I was at first unsure if we were supposed to make a logo since it is supposed to be SpaceX's microsite. In the end I decided that I wanted to have a logo there since I wanted a simple intuitive way of getting the user back to the homepage if they wanted to go back, and also using SpaceX's own logo would be risky when sharing the site online since I don't know if we are allowed to use the logo. I also felt that a logo could be used in promoting the mission to mars and therefore it would have its own logo.

Typography

For the font I wanted to have font that looked good with the design but also has a high readability as it is important for the user to feel like they can spend a lot of time on the microsite learning about the mission to mars. I decided to go with a sans-serif font named

Teko from google fonts. It is a font created with headlines and other display sized text on screen, which I felt matched my criteria's. The Teko typeface features letterforms with low stroke contrast, square proportions and a structure that appears visually simple. Which is I found was fitting to my design and also for the users readability (https://fonts.google.com/specimen/Teko).

Gantt Chart

The Gantt chart was created in week 1 in accordance to part 1 of the exam. Overall the plan was very helpful men trying to keep track of my progress as I worked over the course of 5 weeks. It can get overwhelming to work a large project by yourself, when you are not sure how far along you should be at any given time. I added images of my Gantt Chart and also added a pdf copy in my zip file in case it is hard to read in this document.

SpaceX microsite mission to mars

13-May-2019

SpaceX

Project manager
Project dates 13-May-2019 - 18-Jun-2019

Completion 0%
Tasks 24
Resources 0

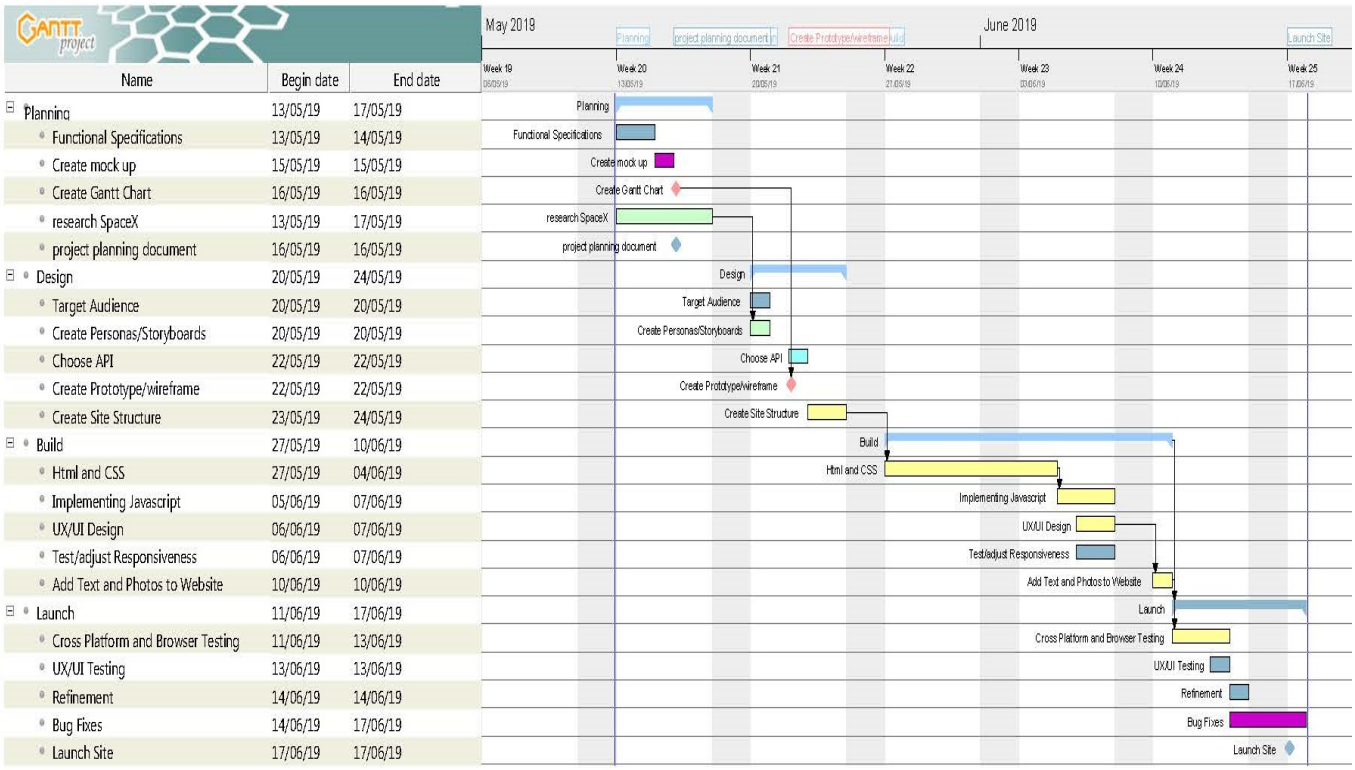
Build a self-contained site for SpaceX about the missions to mars program activity on its own domain.

Tasks

Name	Begin date	End date
Planning	13/05/19	17/05/19
Functional Specifications	13/05/19	14/05/19
Create mock up	15/05/19	15/05/19
Create Gantt Chart	16/05/19	16/05/19
research SpaceX	13/05/19	17/05/19
project planning document	16/05/19	16/05/19
Design	20/05/19	24/05/19
Target Audience	20/05/19	20/05/19
Create Personas/Storyboards	20/05/19	20/05/19
Choose API	22/05/19	22/05/19
Create Prototype/wireframe	22/05/19	22/05/19
Create Site Structure	23/05/19	24/05/19
Build	27/05/19	10/06/19
Html and CSS	27/05/19	04/06/19
Implementing Javascript	05/06/19	07/06/19
UX/UI Design	06/06/19	07/06/19
Test/adjust Responsiveness	06/06/19	07/06/19
Add Text and Photos to Website	10/06/19	10/06/19
Launch	11/06/19	17/06/19
Cross Platform and Browser Testing	11/06/19	13/06/19
UX/UI Testing	13/06/19	13/06/19
Refinement	14/06/19	14/06/19
Bug Fixes	14/06/19	17/06/19
Launch Site	17/06/19	17/06/19

SpaceX microsite mission to mars

Gantt Chart



Name	Begin date	End date
Planning	13/05/19	17/05/19
Functional Specifications	13/05/19	14/05/19
Create mock up	15/05/19	15/05/19
Create Gantt Chart	16/05/19	16/05/19
research SpaceX	13/05/19	17/05/19
project planning document	16/05/19	16/05/19
Design	20/05/19	24/05/19
Target Audience	20/05/19	20/05/19
Create Personas/Storyboards	20/05/19	20/05/19
Choose API	22/05/19	22/05/19
Create Prototype/wireframe	22/05/19	22/05/19
Create Site Structure	23/05/19	24/05/19
Build	27/05/19	10/06/19
Html and CSS	27/05/19	04/06/19
Implementing Javascript	05/06/19	07/06/19
UX/UI Design	06/06/19	07/06/19
Test/adjust Responsiveness	06/06/19	07/06/19
Add Text and Photos to Website	10/06/19	10/06/19
Launch	11/06/19	17/06/19
Cross Platform and Browser Testing	11/06/19	13/06/19
UX/UI Testing	13/06/19	13/06/19
Refinement	14/06/19	14/06/19
Bug Fixes	14/06/19	17/06/19
Launch Site	17/06/19	17/06/19

SEO/Content Stategy/WCAG

The goal of the site is mainly to inform the user of the mission to mars, therefore there is a limited amount of prompts and calls-to-actions compared to what you might expect from a regular website. The main calls-to-action are the buttons that either prompt the user to join the newsletter or continue reading about the mission to mars. The header also contains the headline “Mission to mars” making life multiplanetary which is the same rhetoric Space X’s uses on its own site. I made the conscious decision to add this on the microsite as well as this is meant to be a page made by SpaceX’s mission and therefore is easier for search engines to see the connection between them and is more likely to be what a user would search for if the wanted to find out more about the mission from SpaceX.

I added clear headlines and sub-headlines in an effort to make it more SEO friendly. As well as the order of the text is placed in an effort to make the text clear and also ad buzzwords like. To make the site more mobile friendly I tried to keep the text short, yet informative. I based this on other microsites I have looked at during my research like https://blackhistoryisnow.com/, which is informative but also has a lot of visual stimulation. I tried to keep my page within this similar layout. A microsite often requires a lot more scrolling than a regular site, I kept this in mind when creating the site, so I added elements

from the other pages as a type of introduction to the other main pages. This way the frontpage doesn't get to cluttered with too much information, but also gives you an easy way to flow through the page.

HTML/CSS: Semantics, structure

I wanted to keep the HTML-elements and CSS classes to the absolute minimum required to create this microsite. This makes it easier to maintain control over the HTML elements and CSS that makes up the page.

I've made sure to use the semantically correct html-tags when possible, and tried to reduce the use of unnecessary div-tags. Divs were mostly used as a generic container/grouping element, when no other semantic html-tag would fit the role.

Each page shares the same header and footer. Both the header and the footer utilizes flexbox to place the content. While the footer goes from a row to a column on smaller devices, the header maintains the row layout. The menu-button and logo uses absolute position, to make sure they stay put in their respective corners, no matter the layout.

The pages also share the same hero-image landmark, that contains the mission statement, and a button that makes the newsletter pop-up appear.

The index, lifeonmars, launch and transportation html pages all share the same two column layout. A timeline in the left column, and the page specific content placed in the right column. While the contact.html page only has a contact form centered in the middle of the page.

The two column layout uses flexbox to place the content, and changes to a one column layout on smaller devices, to make more space for the page content.

The timeline uses CSS-selectors that alternates between the timeline-items that are odd, and even to place them on either the left or the right side of the timeline on desktop. If the user views the microsite on smaller laptop and downwards in device size, the timeline will then change its layout to place all the timeline-items at the center of the timeline, to better make use of the space.

When it came to responsiveness, I went for multiple media queries for each element, and the breakpoint were set depending on where the content of the element started to look bad on smaller screen sizes. This makes for a more dynamic responsive layout, since not all the content looked bad on the same screen sizes, so the same breakpoint for I.E. the timeline, would not necessarily translate well to the breakpoint for text or a menu, the content would either bee to stretched or to squished.

I tried to make the CSS classes as modular and reusable as possible, since this is microsite with minimal content. This also makes the CSS easier to maintain, and responsiveness easier to create.

JavaScript

I've used JavaScript to validate user input, make the newsletter modal, pop up when the user clicks the join the newsletter button, and to get relevant data from the api, to be used in the footer and on the pages index and launch.

I have split the JavaScript between 7 different js-files. To reduce the total amount of code in each file, and to make it possible to only load the necessary JavaScript, depending on which page you are on.

Common.js

The idea behind this file is that it contains the JavaScript that is common for all the pages. I decided to put the newsletter.js and validation.js in their own files, although they are commonly shared scripts, to reduce the amount of code in one file, which makes a little difficult to follow.

The common.js file contains a simple fetch method, that retrieves the company info from the company info api-endpoint (https://api.spacexdata.com/v3/info), and passes it down to the createCompanyInfoMarkup-function. This function takes the data retrieved from the api, and creates the company-links html-markup in JavaScript, and then appends it to the correct position in the footer.

Lastly in this file, we have the onMenuButtonClick function, that toggles the main navigation menu on and off, on smaller devices.

Index.js

This file contains all the necessary JavaScript to make the index.html page function correctly.

It starts with a fetch-method, that retrieves information on the next planned launch from SpaceX, by accessing the next launch API (https://api.spacexdata.com/v3/launches/next).

It then feeds the data down to the createLaunchMarkup function. This function creates the html-markup that makes up the .launch section element, with JavaScript.

The function also conditionally checks if the data passed down to the function contains date, rocket and launchsite data, and then displays them if the data does exist. The display conditions were written with ternary if else operators, to save space, and to make it work inside the template string that makes up the markup.

If a date does exist, the function then calls on the getDateTime and formateDate functions. getDateTime formats the date to fit the time-elements datetime attribute format, while the

formateDate formats the date to a format readable for humans. Both the functions utilizes regex to do the formatting

Lastly when all the data has been passed down and formatted, the function then returns the markup, and it gets appended to the api-container on the index.html page.

Launch.js

This file contains the JavaScript for the launch.html page.

It does basically the same thing as index.js. The only difference is that the launch.js chains three API-calls together with the then functions. It gets the data for:

- Next launch(https://api.spacexdata.com/v3/launches/next),
- Upcoming launch(https://api.spacexdata.com/v3/launches/upcoming)
- Past launch(https://api.spacexdata.com/v3/launches/past)

And that creates the markup and appends them in that order. I had to chain the fetch-calls after each other like that, or else the markup would have appended in a random order each time you loaded the launch.html page.

Validation.js

This file contain JavaScript that validate user inputs in forms. I decided to set the novalidate attribute on the forms, so I could get full control over the input validation from my javsacript files. I decided to put all the form validation functions in one master file, and include the script file above the newsletter.js and contact.js so they both can use the validation functions. This helps by making the validation.js file easy to read, and I keeps from duplicating the validation functions, as would have happened if I had included them in both the newsletter.js and contact.js files.

There is a validationMessages object at the top of the page, which contains the messages to be displayed on the form, if the input is not valid. The object property names are the same as the form element names, to make for a dynamic selection of messages, depending on the input being validated.

The scripts start by getting the form on the page, and then adds a click event listener to the form, which takes the validateForm function.

The validateForm functions starts by calling the preventDefault function on the event object, to prevent the page from reloading when the user hits submit. It then loops through all the form inputs, stored in the inputs variable, which retrieves all the form inputs I want to validate, with the getFormInputs function. Each loop begins with storing the current input element insde the input variable. The script then validates the input using the validateInput master/controller function, and then displays or removes the validation message for the

current input, depending by if the input was valid or not. The validation message is retrieved from the validationMessages object, and chosen depending on the input.name shows up above the input-element on form, with an appropriate message. At the end, the function either returns true or false, depending on if the all the form input values were valid or not.

The file contains a getFormInputs function that gets all the form input elements, and textarea, which makes up all the elements inside the form that can take user input. The function works by looping through all the elements inside the form, and the pushes the elements which can take user input, into a new array, that the function returns after the loop is finished

I have three validation functions, and one master/controller function called validateInput.

validateInput contains a switch statements, which takes the input.name as a parameter, and then decides which input to use depending on the input.name.

validateText, that simply checks if the text input is empty or not.

validatePhone, which checks if the phone input starts with one or no +-symbol followed by a combination of numbers of any length.

validateEmail which checks if the input contains a valid email I.E. an email that starts with any combination of letters and numbers and the three special letters(- _ .), followed by an @ followed by the combination as the beginning of the email, and ending with an . and a combination of letters of any length.

All the validation methods utilizes regex, to match the input against a regex-pattern.

Contact.js

This file starts out by retrieving the contact form by its id. Then I add an submit event listener and attach the validateForm function to the listener. If the validateForm method returns true, the form will submit, and the page will reload, to let the user know that the data were submitted.

Newsletter.js

This file starts out similarly to contact.js This file also contains all the functions necessary to create the newsletter popup.

The first function being called on in this file is the getNewsLetterFormMarkup function and stores the returned form element in the newsletterForm variabel. This functions creates a form element, assigns it the appropriate id, and sets the novalidate to true, to enable full control over the user input with javascript, and then sets the innerHtml of the form to the that of the newsletter form, by using multiline template string. Lastly the function returns the newly created form element.

Then the script goes on to createing the modal element, which acts as the forms container, and adds the transparent background that is present while the newsletter is open. After its creating the script appends the newly created newsletter form element to the modal.

After this, the script assigns the validateForm function to the newsletters to the newsletterForm submit event listener. If all the user input is valid the script goes on to close the modal with the closeModal function, and then calls on the howSubscriptionMessage function.

The howSubscriptionMessage is a function that lets the user know that they have subscribed to the newsletter, by showing them a message on screen. The function simply creates an p-element with the subscription message class, which makes the takes up the whole page with and transparent black background, with a large text, centered on the page, in a fixed position. The function inserts the message at the beginning of the body, element, and then removes it after 2.5 seconds.

The script also retrieves the newsletter-button, located in the header, by its id, and attaches an click event listener to it, which takes the openModal function.

The openModal function sets the body overflow to hidden programaticly in javascript, to prevent the user from being able to scroll while the window is open. It also takes the modal element created at the beginning of the script, and inserts it at the beginning of the body element.

Similarly I've attached the closeModal function to the newsletter forms cancel buttons click event listener. This function removes the style attribute from the body, to remove the overflow hidden inline styling that prevents the user from scrolling, and then it removes the modal from the body element.

Slides.js

This file contains the logic necessary to make the image slider work.

I decided to store all the slide data, E.G. image-path, alt-text, slide-title and slide text in an array, congaing objects, representing each slide. This makes it easier to dynamically swap images, instead of having all the slides as html-markup on the page itself.

The script starts off with initializing the slider thumbnails, using the initThumbnails function. This function loops through the slideImages array, and creates the thumbnail img-elements, and then appends it to the thumbnails div-element that functions as the thumbnail container. The thumbnails also gets an click eventlistner assigned to them, and the listener takes in a function that removes the active class from the previously active slide thumbnail, and adds the active class to the currently clicked thumbnail, and lastly the function calls on the getSlideImage. The active class dictates the opacity on the thumbnail img-elements. The ones with the active class have no opacity.

It then goes on to displaying the first slideImage in the array by calling on the getSlideImage function, taking the first slide I.E. slideImages[0] as a parameter.

getSlideImage takes an object as a parameter. The functions start out by checking if the slider already contains an image, and if it does, it removes it.

It then gets the slider image element with the getSlideImagElement method. This method takes the slideImage object as a parameter, and assigns the data to the correct attributes, before returning the newly created img element, so the script can appends it to the .slider figure-element on the page.

The script goes on to setting the slider title, by assigning the sliderTitle.innerText value to be same as the current slide object. It then checks to see if the slider contains any text, if it does it gets removed and replaced with the text of the current slide object.

I decided to put the slider text inside an array, to easier separate multiple paragraphs with JavaScript, each array item represents a paragraph/p-element. The function loops through the text array and creates a slider text paragraph of each text item inside the array.

Lastly the function updates the slider number iterator, located in the top left of the slider by retrieving the index of the currentSlide and adding it with 1 to get the correct number, since the array index is zero based. Putting this number together with the total amount of slides in the array makes out the slider number.

Lastly the script adds a click eventlistner to the next and previous buttons.

The next button checks if the slideIndex is not the same as the lastSlideIndex, if its not then it takes the slideIndex and adds it by one, else it sets the slideIndex to zero, since it means that the slideIndex is at the last slide

The previous button checks if the slideIndex is not 0 I.E. at the first slide. If it's not then it takes the slideIndex and subtracts it by 1, else it sets the slideIndex to be the same as the lastSlide.

Both functions ends by calling on the getSlideImage function and passes the current slide I.E. slideImages[slideIndex] down as the parameter, to get the current and correct slide.

API's

Figuring out how to make the information in the API's fit to the theme of the microsite proved to be harder than I first imagined. Although there were a lot of options, it was hard to figure out how it would fit the overall theme. Therefore I decided to add a Launch overview site that would list recent and planned launches Space X's is planning. The launches are an important part of the work related to developing the technology needed for the mission to mars so the launch page gives an overview of the project as it is today and the progress being made. When the time comes for the launches related to the mission to mars they could also be shown on this site.

I also took use of the Company Info api-endpoint, to populate the footer with relevant company info, like a link to SpaceX official twitter and Flickr account, and also a link to their main corporate home page.

Implementation/rollout

The only implementation this microsite contains is the SpaceX api. It's implemented in index.js and launch.js files. The implementation is used by calling on the fetch method, to retrieve the necessary data from the respective SpaceX url api-endpoints, used on this page. The data is then passed down to the functions that I've made, which creates the html-markup used to display the data, in JavaScript, and then appended to the correct html-elements on the microsite.

I guess you could say I've also implemented GIT for versioning control of the source code.

The microsite will be rolled out and deployed with FileZilla, and all the files that makes up the page will be transferred to my domain(holdcroftkari.com).

Conclusion

For this project the main goal is to create a microsite that focuses on SpaceXs mission to mars program. I wanted to create a microsite that was informative and also visual so that the text wouldn't be overwhelming for the user. I wanted the site to encourage further reading as well as keeping up to date on the program. Since microsites are often build around a certain event, I wanted created a call-to-action to guide the users towards the newsletter that will keep them updated on the project. I wanted to make a microsite that was informative, yet sleek and easy to read. I wanted to keep my text informative, yet to the point and not to cluttered as well as adding images and video to supplement this. The site is meant to be a one stop for information about the mission to mars, and what life on mars could look like in the near future. There isn't much we know for sure about this at this point in time, but the illustrations and video from SpaceX's galleries help paint a picture of what the future of this mission could hold.

References

- Last Read: 2019.05.26, *Microsite.*: https://boagworld.com/marketing/microsite/
- Last Read: 2019.05.25, *About SpaceX.:* https://www.spacex.com/about
- Last Read: 2019.05.25, *The Microsite: A Definitive How-to Guide for Their Effective Use.*: https://boagworld.com/marketing/microsite/.
- Last read: 2019.06.12, How and When To Create a Website With a Black Background Design: https://dev.to/neshaz/how-and-when-to-create-a-website-with-a-black-background-design-2a3e.