Report

PROJECT EXAM

FEU, Noroff

14.06.2019

By: Marie Helen Berger

CONTENT LIST

LINKS TO ASSIGNMENT ELEMENTS

Link to the SpaceX microsite:	http://badponydesigns.com/exam/index.html#CurrentlyHappening
Link to the prototype:	https://xd.adobe.com/view/f6b42a5c-d0dd-4492-68ca-3de1aa5e4567-8085/screen/627eab55-f350-48e3-bbbd-e8147242fac5/MAIN
Link to new, (clean)GitHub repository	https://github.com/mie91/2019-06-16_project_exam_repository_clean_version_mhb
Link to original(messy) GitHub repository *(see the "after_creating_historic_timeline" branch)*	https://github.com/mie91/projectexam_spring2019_marie_helen_berger-/tree/after_creating_historic_timeline

THE REPORT

In this years project exam we were to create a micro-site for SpaceX or NASA to raise awareness about their space programs, goals and activities. This micro-site should have a specific target audience in mind and provide links to further information about SpaceX or NASA. The requirements included a minimum of four pages, be responsive, employ JavaScript/JSON API and some kind of calendar or timeline among other things.

We were given five weeks to complete this exam, and had two mandatory deliveries in week one and two.

THE PROCESS

Week 1:

The first thing I did when we were handed out the assignment were to decide once and for all if it was SpaceX or NASA I wanted to create a site for. I made some research and thought about pros and cons of both of them. Even tough NASA had some very interesting API's and other resources I felt the subject was a little bit overwhelming. SpaceX is a little more “narrow” subject and also had some good resources available. I am also a fan of Tesla and Elon Musk, so I found SpaceX the most interesting one of the two, and decided to go for it. I investigated further which API's I wanted to use and what photo resources were available. I also looked at a lot of different micro-sites for inspiration and to understand what makes a micro-site so special. Most of the examples showed really fancy animations and visuals that would have been awesome to use. But I knew my coding skills at the moment would not be good enough to create something like that. So I had to think a in a more simple and less fancier way. But I can see now that my ideas and plans for the site functions were a little too ambitious (I will elaborate more on this later in the report). But one new thing I wanted to try create was a nice looking timeline that would employ a JSON API.

In week one we were also to deliver a Project Planning Document, Gantt Chart and Functional Specification. In the planning document I included nine milestones and what I believed to be the most critical/risky stages:

- Color-scheme decision
- Implementation of Javascript code

Of all subjects we have worked with throughout this course, JavaScript has been the most difficult for me, and in previous projects, the color-scheme planning has been the most indecisive and time consuming stages. This time however, I decided later that the micro-site should have some familiar roots back to SpaceX's main website design-wise. This is so that they have some connection to each other, and the users would not have to readjust a lot when continue their reading and research when moving over to the main website. So I wanted to keep a color scheme similar to the main site, but with some original elements off course. Writing the Functional Specification was a little difficult as I didn't have my ideas for the design and functionality 100% clear at the time. But It has been updated at a later time (both versions are included in the .zip delivery).

Week 2:

In week two we were to decide for a target audience, write about various personas, storyboards and create a prototype/wire-frame. I started with deciding a target audience, but found this surprisingly difficult as I felt like this page and it's “mission” would probably fit most users. So I had problems with narrowing and single out a defined target audience, and ended up with a to big and broad audience. I also received feedback on this exact thing from our teacher *after* week two. So I made some adjustments to the target audience and define them now as:

> *“My idea for the target audience is mainly adults/young adults/students who are looking for an introduction to SpaceX, and the micro site will serve as a portal to “SpaceX knowledge”. They might not be familiar with the company at all, but are curious about their goals and missions.*
>
> *They also want to get the information in a entertaining and engaging way. The technical skill level will probably vary, so therefore it is also important that the page is*

easy to navigate through, and user friendly. The information will also be basic and "easy-to-get", not very technical and heavy. The audience will however be directed to the main website should they want to delve deeper into SpaceX."

Because of this change in target audience, I also had update the personas a little bit as well (as with the Functional Specification, both versions will be available in the .zip folder).

We were also to create a prototype/wire-frame for the page this week. I decided to create it in Adobe XD as I am really fond of working in XD and like to have a detailed prototype to use as a reference. When I made my early research for ideas and inspiration for page design, I stumbled upon examples of the "parallax scrolling effect"(PSE). This is a effect "*where the background content (i.e. an image) is moved at a different speed than the foreground content while scrolling*". I really wanted to try to get it to work on my project, and it became a foundation for my page design. In addition to this, I wanted to include a timeline as stated earlier, and have a engaging, eye-catching website.

When it came to the user experience and interaction design I started with creating a kind-of-card-sort, where I made list of what pages I wanted to include and their content (API's, facts, functions etc.). I then sorted them after how I thought it would be most logical. Then I made my boyfriend and a other friend sort them after their own preferences (I explained the concept of micro-pages beforehand). Then I compared them to each other. They were quite similar all three, with some minor differences. The biggest differences from my own was that they both was unsure what to include as a navigation bar, and listed everything on one page. This was because they had some difficulties to understand how this micro-site would end up different from a "normal" webpage, especially if it contains a regular navigation bar with four links, and footer menu . The reason for this is that they both (and me, before reading the definition of a micro-site) associate micro-sites with a single, long page with no navigation other than scrolling downwards. This is not a requirement for a micro-site, regardless I am still a little unsure if I have managed to create a website that would classify 100% as a micro-site.

Even though I did not receive a clear navbar preference with my two test subjects, I got a indication on how they would sort the content. But in hindsight I should have included even more API's, subject options and content, even the things I did not find interesting to include

at all, as the other users might have thought otherwise.

In regards to scrolling, I am usually not very fond of pages where I have to scroll a lot to get to where I want. My test subjects in former projects have also expressed the same opinion. But at the same time, I believe it might depend on what type of page you are on. If you have to do a lot of jumping between pages in combination with a lot of scrolling, it can become really bothersome, fast. Are you just reading and making research it might be more comfortable being able to just scroll downwards as you read, especially if you are on mobile devices. So my idea here was to have quite long pages, but also have a fixed navigation bar with drop-down menus, so that the users would always have the option to jump back/forwards to wherever they want, but also be able to scroll and just “surf” and scan the page if they want.

In addition to this navigation bar, I would also include some collapsible content divs. The idea behind these are that some of the users might be “regulars” and want to scan the page fast looking for specific things (such as the “next launch” API), and don't want to “jump around” using the navbar. So some less important content, or very long tables etc. is hidden at default and the users have to click to expand the area. This also makes the sites not excessively long.

The assignment also specified that the micro-site should provide links to further information about SpaceX. So I also wanted to include a visible link to the main website in the navbar, and also have a lot of links to further information spread around the site. These links should also be opened in a new window, so that the users would not be removed from the site, and could easily return to their exploration on the micro-site when closing down the wikipedia page, etc. SpaceX is a private company and does not have to comply with the government web guidelines, which include WCAG 2 AA, as there are no requirements for non-government websites. But they are deeply connected with NASA and probably have a lot of the same users and visitors, so I wanted to try keeping a AA rating throughout the page anyways.

The prototype has a lot of different colors (inspired by their header image) and I used a lot of linear gradient effects on the background, but this made the site design look somewhat disjointed and inconsistent (the feedback I received later from the teacher also said this). So later I changed and simplified the color-palette. I somehow always end up with dark themed page-design, but in this specific case, I thought it might work quite well.

I ended up with a page with dark backgrounds and white text. I also added some deep-red buttons to give some color to the otherwise grayscale themed design. In regards to font choice, I usually always use a combination of two fonts. But I found “Roboto” on googlefonts and really liked it. Roboto is a is a neo-grotesque sans-serif typeface. It looked very good as headers with letterspacing and as basic paragraphs. So I decided to only use this font.

Week 3-5

When the week two delivery was completed I started right away with the preparations for the coding and site building. I made some of my friends test the prototype. They all said they liked the design, but did unfortunately not provide a lot of input. Then I wanted to make sure I was able to create the parallax scrolling effect, as it was a major element in my design. But luckily, if I was to be to difficult for me and my coding skills, the prototype and design idea would still work, as the background-images would just be regular, fixed images. I was not able to create a “100% real” PSE as it is dependent on the use of JavaScript. If I was to use this, I would be totally dependent on tutorials and “how-to's” and still not *really* understand what I was doing. And as our teachers have stated a lot of times, just following a recipe to the letter is not what we are supposed to do. So I looked in to some other ways to do this, without any use of JavaScript. I also made sure I understood every action and why/how it worked before I used them. I ended up with a middle ground between a fixed, normal image and a PSE. I only used CSS and HTML. It does not look as cool as the real deal, but it's still nice in my opinion. It was the same deal with the sticky navbar and dropdown menu. I have not created a navbar like that before, so I had too look at some examples at WC3 Schools and other pages. There were some help in these off course, but I had to make a lot of adjustments and changes to get it to work and look like I wanted. I got some feedback on my footer in the prototype, saying it had too much happening on it and needed to be a little more simple. I do agree on that, and made a very basic, simple footer including the normal social media links etc.

The coding process of the four pages

Index Page

The index/start page is called “Current” and contains firstly general and current info about Space X as a company,a short introduction and a newsletter sign up form. From there it goes to “Next launch” where there is information on when and where the next launch will be, and a countdown to the launch date. Next there is information on the various rocket types. Lastly there is information and overview of the launch pads that SpaceX use.

The index page was the first page I created and is the start page of the website. It worked as a template for the three other pages in some degree. I originally planned to have a start page that only had a full-screen video background of a launch etc. where you only had a button that said “explore” and then it would direct the user to the “Current” page. The idea was that it would build up excitement for the user. But I decided to leave this idea behind, as one of my test subjects (and other testers later when I asked their opinion) made a good point, saying it was is kind of pointless and would probably annoy them if they were to re-visit the site at some point, and/or use it regularly.

So the first thing the user will see is the “Discover SpaceX” header and photo, and get some general info about the company. Here I wrote the first API fetch in my JavaScript. I remembered to really struggle with it in the JavaScript CA earlier this year. And thought it would be a nightmare to get it to work. But to my surprise, I got it to work on the first try, and felt that I got a better understanding of the whole JSON API stuff when re-visiting this. I used the innerHTML rather than the createElement function as I am more comfortable using it(I know the createElement function is a lot “cleaner”, so in the future I will probably try using that instead). I also included a little thumbnail gallery that opens a new window with a bigger photo when clicked. Ideally I would like to create a image carousel and being able to enlarge the photo on the same page as the thumbnails. But this was beyond my JavaScript skills as well. Under the info part I made a collapsible div that contains a sign up for a newsletter. This will expand when clicking the arrow icon, and there is a header indicating that there is a way to receive updates by clicking this icon. SpaceX is probably not that dependent on “selling” and persuading the users the idea of newsletter-sign-up as shops and other online services. Therefore I thought it would be better having it not taking up a lot of space and being very “in your face”. I am also quite happy with how the collapsible div turned out. I have some experience with it now, as I have used it a couple of

times in previous assignments.

Next we have the “Next Launch” part. Here there is a JSON fetched that provide information on the next launch. I also wanted to include a countdown as it's very space-rocket-y (and perhaps very cliché..). I must admit that I used a WC3 Schools a lot when creating the countdown. But I was not able to find any other way. But I tried to get an understating of the function and how it worked. I also tried in a very ambitious and (very)futile attempt to connect the countdown to the API, so that the countdown would also update when the launch info would. But I did not get it to work. So the countdown function will quite possible be either showing the wrong date or show “LAUNCH DATE HAS EXPIRED” when the teacher is evaluating my exam. The same goes for the map showing the location.

After this we have the “Rockets” section. Here I included a very nice illustration of the SpaceX rocket family that a fellow student shared on the Noroff Discord server. In the collapsible I added the JSON providing the rocket information. The reason for not having the info above, was that people might first be only interested in how the rockets might look. Then they could choose to see a table with information. At this point I had also inspected the Launch Pad JSON and saw that on *status:* it said “active/inactive” rather than “true/false” as the rocket JSON. This irked me quite a lot, so I made a function that changed the words true/false to inactive/active. The idea was that the words would also be colored green/red. But I was not able to make the latter work. In regards to media queries; most of the horizontal tables and timelines where adjusted so that they are vertically lined when displayed on screens smaller than a regular-sized tablet.

Lastly we have the “Launch Pad” section. This section follows a lot of the same recipe as the “Rocket” section. Here the table is vertical as there was a lot more launch pads than rockets. I also included a iframe with a youtube video, as I felt like I needed something more than images. It also makes the user aware of all the videos SpaceX has available on their YouTube channel. This iframe did however cause some errors in the console command. I tried to investigate, but I am still not sure what they mean. But they don't seem to have any negative effect on the page.

Contact Page

The contact page is a very simple page, with the usual form input. I only included name and email address in addition to the text field, as there is probably no need for street address or phonenumber, when sending questions etc. to SpaceX from this page. I did also include a checkbox for the newsletter under. So that you could sign up at the same time. The JavaScript validation coding went quite ok. This is probably one of the things we have worked the most with in JavaScript. I had some issues at first with getting the input error notifications showing up when clicking submit. This was because because of some error with #id in the html. I also used the regextester.com to test my regEx code.

Under the contact form in a collapsible div I added some SpaceX headquarters location information, and a list of links to where you could reach SpaceX on social media. The social media part is probably a bit redundant as the same links are below in the footer. But it is also OK I thought to gather all the links in one place, together with all the other contact info.

Future Page

The “Future” page consist of a timeline showing SpaceX's future plans and goals and a table showing info on the upcoming launches and their date and location.

As with a lot of my plans for the page-design, I had to downgrade my ambitions in regards to the timeline as well. There was so many cool, beautiful looking timeline examples on the web, but none were really on my skill level. But I found some that were a somewhat less technical and used some elements from them to get the idea of how I could create a cool looking timeline that “jumps” from left to right. I saw some examples of how to add animation with only use of CSS, and read a lot about it and experimented a bit. But decided to play it safe and keep it simple. So there was no fancy animation this time. (The information I used on the Future-timeline I gathered from articles from BuisnessInsider.com and Inverse.com. I did not have the time to write the sections in my own words unfortunately, and I hope that it was OK to do, as long as I refer to the articles in my resource list.)

Under the timeline (which I also kept inside a collapsible div as it was quite long) I fetched the JSON for upcoming launches and was able to display it in a big table. I had big problems getting the this table to look good on small screens, so here I also had to do a lot of Googling, and found some tips showing how you could make the table display in vertical, separate boxes using the ":before" and "nth-of-type" CSS tags. I didn't know a lot about these tags from earlier, but found them very helpful when creating the other timeline on the "History" page.

History Page

This page was the last one, and also probably the one I struggled the most with. Here I wanted the "History" JSON to be displayed in a similar timeline as the one on the "Future" page. But I could not get it to display correctly for a long time. I started from scratch probably three or four times. The problem was getting every other div from the innerHTML float to the right/left. First I made two Fetch request in JavaScript, one fetching the odd numbered objects and place them in a div called "right-something", and then create a second fetch request fetching only even numbered objects and place them in "left-something" div. But no matter what I did, it would not work. And I almost gave up the whole timeline look. But then I made the fetch request gather all the objects again and then rather use the ntchild selector I had learned about when researching the "nth-of-type" tag earlier. Then I was able to select odd/even using these CSS tags and give the left/right text boxes their own properties imitating the timeline on the "Future" page. I had to make some not-ideal solutions that is probably bad semantic vise; such as the use of "direction:rtl". This makes the left textbox look correct, but it also makes the direction of the text right-to-left. So the periods for example are on the start, in front of a new sentence. I tried to fix this, but was not able to do so unfortunately. Also, in yet another hindsight; I should have made the newest history information box first/on top, or made a link on the top to get to it at once if you use this site a lot.

TESTING/IMPLEMENTATION/ROLLOUT

When I finally got my site to look good in every screensize example in Chrome Dev. Tools I did the first upload to my webhost One.com. To my horror nothing worked as it should. No JavaScript, images, nothing! It didn't help that my web connection were horrible or that One.com had a bad day as well. After some (horrible) time though, I found the problem. I had “src:”./images/filename”” etc. when I should only have “src:”images/filename” instead. The reason for this is that I am so used to needing the “./” as my file directory usually looks like this: “index.html>caseAssignment>CourseName” but here it was “index.html>exam”. When correcting this, it all worked again, phew!

I then tested it on my own phone and other PC, a long with my boyfriends phone and PC. Then next day I got my friend that also is a student on this course to test it very thoroughly on her devices. It all looked like it worked fine. The only thing she noticed was that the font-weight were thinner in Firefox than in Chrome. I was also very lucky as my parents where having a big gathering at their house yesterday, so my parents (60 years, OK with computers) , siblings (30-40 years, various computer skills) and nieces (15-18 years, good computer skills) that were present there also tested on their phones. And it seemed like everything looked and worked ok on their devices as well!

EVALUATION AND CONCLUSION

I have a lot of mixed feelings in regards to this assignment and the result. I can start to evaluate what I am pleased with first. I am satisfied with my JavaScript in general. I haven't really done anything else than what we have already worked with earlier in the course, but I really had a hard time understanding it. So I dreaded to start working with it, and expected to have a heart attack and probably throw my computer out of the window in the process. But to my surprise, it went quite well. I felt like I could look at it with new eyes, but still have the basic knowledge from earlier helping me to get a grasp on it. And the fact that I in some degree could decide how much I would use it, and what to use it for helped a lot on my motivation unlike having strict requirements as to what I should be able to do. I'm still quite terrible at it, but I will work trough the summer with the basics and prepare for Javascript 2 in the autumn. I am also satisfied with the parallax part of the design. A lot of

the testers commented that it looked great and “made” the site. It did however not look optimal on some phone screens. And my plan was to resize the images and optimize them for smaller devices, but there were unfortunately not enough time. I also like the sticky navigation bar. It makes it a lot less tedious navigating trough the site, and it also works quite well on small devices. The only thing that **really** irks me with it, is that I was not able to center align the buttons on small devices. This is probably because there are actually two types of buttons; drop-downs and not drop-downs. If I had more time this would be one of my top priorities to fix, as it really ruin the look. I also like both of the timelines, even though I would have loved having them animated as the user scrolls downwards. They are a little boring right now I feel, but I am also happy that I was able to make the history one a “JSON timeline”. I have also tried to make sure to add “alt=text” to every image, and I do believe the color scheme meets the WCAG AA requirements over all. I also started out good with using <article> and <section> tags, but forogt this on the way, and it is probably not optimal now.

Then over to what I am not so happy with. First of, I don't think the color scheme is optimal. I liked it in the start, but now I am not so sure anymore. I would also have liked to use more linear gradient as background (gray-scale) as it makes the page look a little more exciting and dynamic. Because right now after analyzing and spending five weeks with it, I find the whole color scheme quite boring and uninspiring, and I'm not sure the red colored details are correct either. This might be because I have worked a lot with it and is just tired of it. My testers and others have complemented it or been neutral. Nobody has uttered any negativity yet, at least. But then again, most of them don't really have great knowledge about web design either.

There is also small thing I have forgot to include, such as a link on “Next Launch” to “Upcoming launches”. Also I should've probably added a arrow icon pointing upwards and a function hiding the collapsible again. But at the time I thought the user could just use the drop-down menu instead. But some of the testers said it would still be nice to have.

I also regret not having more API's and general info and pictures on the site. I feel that it's a little empty.

Then there is my code in general. Every time I start a new project/assignment I tell myself “*this time the code will be super clean, tidy and organized*” . No “div class=”divOfTheDivTwoThree”. But alas, it was not to be this time either. It always start out

quite good, and by the end it's complete chaos. There is probably some CSS tags/properties that don't even serve a purpose and that does not function at all, but I am to scared removing anything, thinking it will make all collapse.

Then it is the GitHub part. When I read the assignment I understood that we were just to upload the finished project here as well, like we have done in other assignments. Then a fellow student told me in the middle of the project that we should probably pull, commit etc. as well. So I had to read up on the git part again, as I had forgot a lot about how to use it. But in the end I have not been able to use it correctly. I was able to do it right as long as I just updated the master branch. But as soon as I started experimenting with branches, it became chaos. I lost progress, files etc. on multiple occasions. And became afraid to use it really (“luckely”, throughout the exam period I have had some web connectivity issues as well as power outages due to roadworks where I live. So I have made my own back-up system in addition to Git) I should definetly read more about it, and gained more knowledge before using it. So the first, original repository is a mess, that makes no sense what so ever. The branch “*after_creating_historic_timeline”* looks like it might be correct, but I am not completely sure at this time.Therefore I will also try to upload the finished result in a new repository as well, and link to both of them here in the report.

There is also probably a lot of other things and details that will haunt me after the delivery that I forgot to mention here. And to conclude this report; there is a lot of things I would have done differently next time, especially when it comes to the color scheme and use of pictures. But I have worked really hard on it, and even though I felt the creative part of me failed this time, I did however learn a lot of new stuff, especially in CSS, and I got a tiny bit more confidence in regard to JavaScript. It is really fun when it works, and I am really motivated to get better and be able to write code that creates the awesome effects and functions that I was not able to use in this assignment!

Project Exam

Target audience, Personae/Storyboards and Wireframe/Prototype

version 2.0

By: Marie Helen Berger

15.06.2019

TARGET AUDIENCE

One of the microsite's goals would not only be to wake peoples curiosity about space travel and SpaceX but also try to make the subject fun and engaging to read about.

> *“My idea for the target audience is mainly adults/young adults/students who are looking for an introduction to SpaceX, and the micro site will serve as a portal to “SpaceX knowledge”. They might not be familiar with the company at all, but are curious about their goals and missions.*
>
> *They also want to get the information in a entertaining and engaging way. The technical skill level will probably vary, so therefore it is also important that the page is easy to navigate through, and user friendly. The information will also be basic and “easy-to-get”, not very technical and heavy. The audience will however be directed to the main website should they want to delve deeper into SpaceX.”*

PERSONAE/ STORYBOARDS

Leah

"The young student that don't like working on computers."

Name:	Leah
Age:	22
Work:	Student
Technical skills:	Poor
Interests:	Tennis
Goals:	Find facts about Space X and their future missions for an assignment.
Usability needs:	Easy to navigate through the website, and user friendly solutions.
Device:	Computer
Other:	Leah has impaired vision. Her eyes strain when there is to much colors and small text. She is overall not very fond of working on computers, and prefer books.

David

"David is a big fan of SpaceX and wants to attend the next launch located at Cape Canaveral"."

Name:	David
Age:	37
Work:	Engineer
Technical skills:	Excellent
Interests:	Everything futuristic, airplanes and spacetravel.
Goals:	Gain information on when the next launch is, and learn about the mission
Usability needs:	User friendly, easy to navigate to the launch-time table.
Device:	Computer
Other:	Big fan of SpaceX, is a regular user of the site, as he tries to follow the launches closely.

Carol

"The kid dreaming about becoming an astronaut"

Name:	Carol
Age:	20
Work:	Student
Technical skills:	Very good
Interests:	Spacetravel, astro-physics and crossfit.
Goals:	Be kept up to date on when the next Space X launch is. And being able to send her questions to Space X via web.
Usability needs:	Informative, engaging and userfriendly.
Device:	Mobile phone
Other:	Dreams about becoming a astronaut for Space X. She knows what there is to know about their Space Programs, but has some questions on what career path etc. she should choose to reach her goal.

Mark

"SpaceX? What is that?"

Name:	Mark
Age:	32
Work:	Accountant
Technical skills:	Good
Interests:	Paragliding, Hunting and Photography
Goals:	Learn about what SpaceX is, and get an understanding of what they do and what their future plans are, without getting bored while reading about it.
Usability needs:	Informative and entertaining. Responsive, as the user is always on foot and mostly uses his tablet or phone to surf the web.
Device:	Tablet
Other:	People talked about SpaceX at Marks workplace, and he felt hes colleagues thought he was weird that he was completely unfamiliar with SpaceX.

Storyboards

1 - Leah

2- Mark


Did you see the launch?
Yeah! SpaceX is cool
Never?!
Whaaaat?
Mark is hearing talk about SpaceX all the time at work, and is not familiar with the company at all. After hearing how weird it is that he does not know anything about it from his co-workes, he decides to educate himself on the subject (IF its entertaining and fun that is).
They sent a Tesla TO SPACE?!!
While out hunting (Marks favorite hobby) he found some time and used his tablet to read about SpaceX. He found an website that was very entertaining, informative and worked well on his tablet. He became a huge fan of SpaceX and looked forward to discuss SpaceX's next misson with his co-workers.

WIREFRAME/PROTOTYPE

Link to Adobe XD prototype:

- https://xd.adobe.com/view/f6b42a5c-d0dd-4492-68ca-3de1aa5e4567-8085/

General

Notes to the prototype:

- I will try to implement a type of parallax scrolling on the different pages.
- The info and text in the prototype is not the final text, it's only “filling”.
- The NAVbar will maybe be a “sticky” NAVbar.
- Will try to make the timeline animated when scrolling down.
- Not WCAG tested.

Mobile/Tablet (not created yet)

Notes to the mobile/tablet prototype:

- Will keep the parallax effect, if it turns out ok. But will have to make some changes in aspect ratio of the background images.

Project Exam

Project Planning Document
&
Functional Specification Document

version 2.0

By: Marie Helen Berger
19.05.2019

PROJECT PLANNING DOCUMENT

Project start:	13/05/19
Project deadline:	16/06/19
Planned delivery:	14/06/19
Risk/Critical stages:	• Color-scheme decission (30.05) **(moderate)** • Implementation of Javascript code (03.06-07.06) **(high)**
Milestones:	1. Delivery of Functional spec, project planning document, Gantt chart 2. Delivery of Target audience, persona/storyboards, wireframe/prototype 3. Color scheme decission 4. Finish HTML 5. Finish JavaScript 6. Finish CSS 7. Testing with users on various devices 8. Finish report 9. Delivery of report with link to microsite, all files compressed to ZIP

FUNCTIONAL SPECIFICATION DOCUMENT

Date	Version	Document Revision Description	Document Author
19/05/19	V1	Draft Version	Marie Helen Berger
16/06/19	V2	Final Version	Marie Helen Berger

Introduction

SpaceX is a private American aerospace manufacturer and space transportation services company. We are to create a microsite* for them to raise awareness about their space program.
The microsite should appeal to a specific target audience and provide links to more information, live feeds of launches, and so forth.

**Microsite = a branded, self-contained site, usually on a its own domain, with a single purpose and a limited number of pages.*

Functional Specifications

Purpose/Description	Create a microsite for SpaceX to raise awareness about their space program.

• Features	• Responsive design
	• Timeline
	• Link to SpaceX's main site
	• Social Media Links for microsite and mainsite
	• Info boxes with fetched API'S about:
	  ◦ Company Info
	  ◦ Next Launch
	  ◦ Rocket Info
	  ◦ Launch Pad info
	  ◦ Upcoming Launches
	  ◦ Historic events (timeline)
	• Contact form with JavaScript validation
	• Design will meet WCAG requirements
	• Information about the spaceprogram
	• Photos and informational graphics
	• Countdown.
	• Tables with various information
	• Future plans timeline
	• Collapsible text areas
	• Sticky Navbar with dropdown menu
	• Parallax scroll effect

Use Cases

UC-1	Create a microsite for SpaceX to raise awareness about their space program.
Primary Actor(s)	Visitors of the microsite
Trigger	User wants to retrieve information about SpaceX's space program.
Pre-conditions	User is googleing “SpaceX” and got the microsite as a result, and decide to investigate the site.
Post-conditions	The user has found the information they were looking for, and has saved it to his/her bookmarks for future visits.
Main Success Scenario	1. Googles “SpaceX” 2. The microsite is one of the top results 3. The user chooses to investigate the site 4. The user navigate around the site, and find the information he/she was looking for, in addition to other interesting finds.
Extensions	If the users only choose SpaceX's main website when googleing, there will also be links to the microsite here.
Priority	High

The Gantt Chart

The chart is using a hierarhic setup, with eight main-headers:

- Planning Stage 1
- Planning Stage 2
- Design
- HTML
- JavaScript
- CSS
- Testing
- Delivery

All of these headers have multiple tasks under them, written in the order they should be completed by. Some of the tasks will be done simultaneously, as they are connected to each other (in the Design part especially).
Each of the headers and their related tasks have their own color, so that you'll get a better overview of the planning.
I have planned without using the weekends (except this week because of national holidays etc.) , as that would be preferable for my everyday-life. I know by experience that this probably won't be the case, but it is a nice “safety space” to have in the planning regardless.
I also know that the most risky stage in the assignment will be the implementation of JavaScript in the code, as this is the most difficult field for me to work with at this moment. So I have planned the Javascript part between two weekends, so that I have some extra time if needed, and I have no other tasks at the same time. So that I can concentrate fully on the JavaScript coding (*if* I manage to follow the plan, that is). I have also planned to use three days on the report writing. Usually I am able to write a report in a day, especially if I'm good at writing down notes during the whole project. But the report requirements looks to be more comprehensive in this assignment, than the others, so I have given this a bit more time.

Project Exam 2019

19-May-2019

http://

Project manager	
Project dates	13-May-2019 - 15-Jun-2019
Completion	23%
Tasks	39
Resources	0

Gant chart for the project exam of Spring 2019

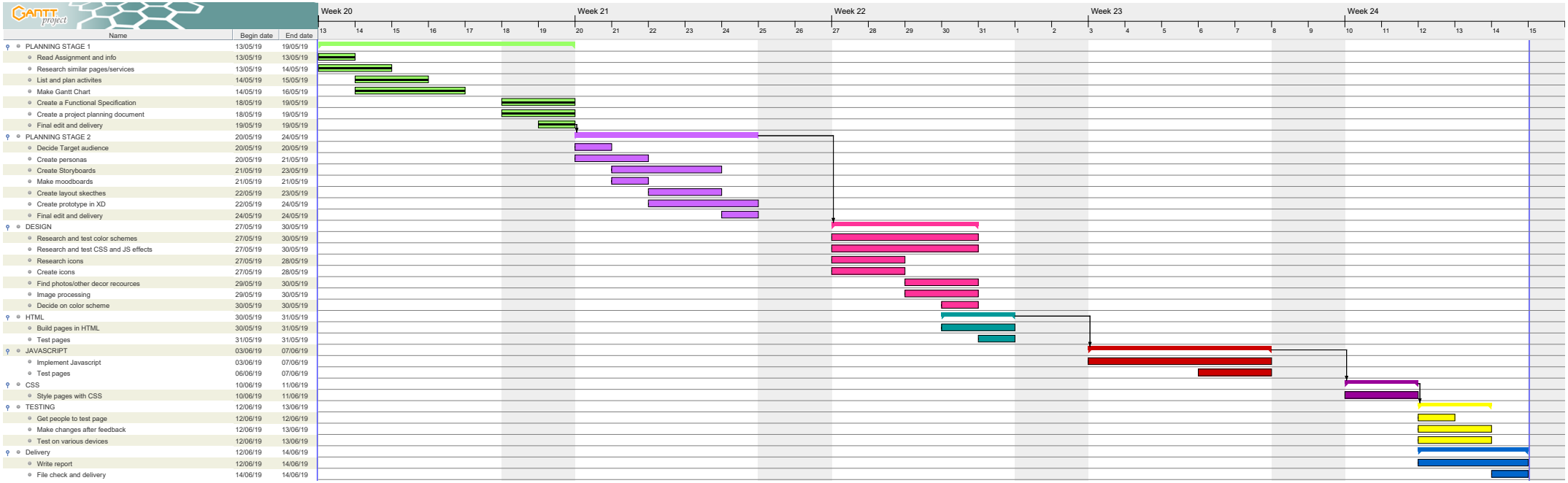
Tasks

Name	Begin date	End date
PLANNING STAGE 1	13/05/19	19/05/19
Read Assignment and info	13/05/19	13/05/19
Research similar pages/services	13/05/19	14/05/19
List and plan activites	14/05/19	15/05/19
Make Gantt Chart	14/05/19	16/05/19
Create a Functional Specification	18/05/19	19/05/19
Create a project planning document	18/05/19	19/05/19
Final edit and delivery	19/05/19	19/05/19
PLANNING STAGE 2	20/05/19	24/05/19
Decide Target audience	20/05/19	20/05/19
Create personas	20/05/19	21/05/19
Create Storyboards	21/05/19	23/05/19
Make moodboards	21/05/19	21/05/19
Create layout skecthes	22/05/19	23/05/19
Create prototype in XD	22/05/19	24/05/19
Final edit and delivery	24/05/19	24/05/19
DESIGN	27/05/19	30/05/19
Research and test color schemes	27/05/19	30/05/19
Research and test CSS and JS effects	27/05/19	30/05/19
Research icons	27/05/19	28/05/19
Create icons	27/05/19	28/05/19
Find photos/other decor recources	29/05/19	30/05/19
Image processing	29/05/19	30/05/19
Decide on color scheme	30/05/19	30/05/19
HTML	30/05/19	31/05/19
Build pages in HTML	30/05/19	31/05/19

Tasks

Name	Begin date	End date
Test pages	31/05/19	31/05/19
JAVASCRIPT	03/06/19	07/06/19
Implement Javascript	03/06/19	07/06/19
Test pages	06/06/19	07/06/19
CSS	10/06/19	11/06/19
Style pages with CSS	10/06/19	11/06/19
TESTING	12/06/19	13/06/19
Get people to test page	12/06/19	12/06/19
Make changes after feedback	12/06/19	13/06/19
Test on various devices	12/06/19	13/06/19
Delivery	12/06/19	14/06/19
Write report	12/06/19	14/06/19
File check and delivery	14/06/19	14/06/19

Gantt Chart



Name	Begin date	End date
PLANNING STAGE 1	13/05/19	19/05/19
Read Assignment and info	13/05/19	13/05/19
Research similar pages/services	13/05/19	14/05/19
List and plan activites	14/05/19	15/05/19
Make Gantt Chart	14/05/19	16/05/19
Create a Functional Specification	18/05/19	19/05/19
Create a project planning document	18/05/19	19/05/19
Final edit and delivery	19/05/19	19/05/19
PLANNING STAGE 2	20/05/19	24/05/19
Decide Target audience	20/05/19	20/05/19
Create personas	20/05/19	21/05/19
Create Storyboards	21/05/19	23/05/19
Make moodboards	21/05/19	21/05/19
Create layout skecthes	22/05/19	23/05/19
Create prototype in XD	22/05/19	24/05/19
Final edit and delivery	24/05/19	24/05/19
DESIGN	27/05/19	30/05/19
Research and test color schemes	27/05/19	30/05/19
Research and test CSS and JS effects	27/05/19	30/05/19
Research icons	27/05/19	28/05/19
Create icons	27/05/19	28/05/19
Find photos/other decor recources	29/05/19	30/05/19
Image processing	29/05/19	30/05/19
Decide on color scheme	30/05/19	30/05/19
HTML	30/05/19	31/05/19
Build pages in HTML	30/05/19	31/05/19
Test pages	31/05/19	31/05/19
JAVASCRIPT	03/06/19	07/06/19
Implement Javascript	03/06/19	07/06/19
Test pages	06/06/19	07/06/19
CSS	10/06/19	11/06/19
Style pages with CSS	10/06/19	11/06/19
TESTING	12/06/19	13/06/19
Get people to test page	12/06/19	12/06/19
Make changes after feedback	12/06/19	13/06/19
Test on various devices	12/06/19	13/06/19
Delivery	12/06/19	14/06/19
Write report	12/06/19	14/06/19
File check and delivery	14/06/19	14/06/19

Content and resource list:

Space X information:

- https://en.wikipedia.org/wiki/SpaceX_launch_facilities
- https://www.businessinsider.com/elon-musk-spacex-mars-plan-timeline-2018-10?r=US&IR=T#2022-launch-two-missions-to-mars-full-of-cargo-and-supplies-but-no-people-5
- https://www.inverse.com/article/51291-spacex-here-s-the-timeline-for-getting-to-mars-and-starting-a-colonyhttps://www.inverse.com/article/51291-spacex-here-s-the-timeline-for-getting-to-mars-and-starting-a-colony
- https://www.spacex.com/
- https://docs.spacexdata.com/?version=latest
- https://www.flickr.com/photos/spacex

Coding recources

- https://css-tricks.com/responsive-data-tables/
- https://flaviocopes.com/css-responsive-table/
- https://www.w3schools.com/howto/howto_js_dropdown.asp
- https://www.w3schools.com/howto/howto_css_parallax.asp
- https://codepen.io/AndreyUgarov/pen/xOrARE
- https://www.w3schools.com/howto/howto_js_navbar_sticky.asp
- https://www.w3schools.com/howto/howto_css_fixed_menu.asp
- https://codepen.io/giannoulakis/pen/XaEZmd

Other (photos, icons etc.)

- https://dribbble.com/shots/1573944-Freebie-iOS7-Style-Social-Media-Icons
- https://www.stickpng.com/search?q=space%20x&page=1
- https://pixabay.com/