# QUASAR

Unix Learning Company

# Introduction

Quasar is a site dedicated to teaching unix language from novice to advanced users. From accessibilty to comprehension, our main goal is to be an educational tool for web development.

Our a team consisting of six members—Matt and Mark, the Content Creators, Zoe, the Graphic Designer, Josh, the head of Interactive Design, Jason, the Information Architect, and Julian, the team leader. We intend any user of our service to be able to become proficient at web development, hosting, maintenance, and management along with many other pieces integral to the operation of a successful enterprise online system.

Alongside the plethora of learning tools, Quasar contains beautifully displayed and easily accessible information about the configuration, integration, and development of Unix systems for beginner to advanced users. Quasar bridges the gap between entry-level and super-user by delivering user-friendly and accessible information for any skill level, this design philosophy allows Quasar to be invaluable for individuals trying to grow their knowledge to helping corporations to ensure that they have the most up to date information to train new members or a reference database to ensure the correct execution of complex operations at critical moments without the need to maintain multiple knowledge-bases.

# Site Goals

## Goal 1

Quasar seeks to provide the user with an easy-to-use user experience that is accessible to all who are looking to learn about Unix

## Goal 2

Quasar seeks to become an educational tool for novice users or professionals to use in order to heighten or advanced their Unix knowledge.

## Goal 3

Quasar provides a unique experience that not only challenges the users knowledge on Unix but allows for easy to remember content as well.

## Goal 4

Give users an experience that will keep them coming back in the future.

## Goal 5

Keep up-to-date content that will continue helping users without becoming outdated.

Quasar seeks to provide the user with an easy-to-use user experience that is accessible to all who are looking to learn about Unix. We want our site to simply and effectively explain Unix as well as its commands in a manner so that anyone can learn about Unix. Our site will provide the user with a well-designed and pleasant-looking website that the user can easily retain information from as well as find what they're looking for.

## Audience Definition

### Persona A

Name: Connor

Age: 20

Occupation: CS Student

Education: Some college

Goal: Pass an introductory web development course.

Summary:
Connor is a college student who just started his first web development course and wants to find more resources to complement his course education and help him pass the class. Connor does not have any web development experience, but he is learning more about it each week from class. Connor is looking for key-words that match topics covered in class and has a foundational understanding while using the website. Connor's goal is that he can gain practical experience using the website.

## Persona B

Name: Steven
Age: 40
Occupation: Web Developer
Education: Bachelors in Web Development
Goal: Find a Unix command he has not used in a while to accomplish his work.

Summary:
Steven is a Web Developer at a professional company and is performing a routine web search to learn more about a Unix command he has not used in a while. While Steven may not remember the exact command name or syntax, he is aware of how it works and what the command can be used for. Steven wants to bookmark our website so that he can return to it whenever he forgets a Unix command or is looking for a refresher on the material. Steven's goal is that he can find the information required on our website to complete his task, and save this resource for future references.

## Persona C

Name: Stephanie

Age: 40

Occupation: Web Development Professor

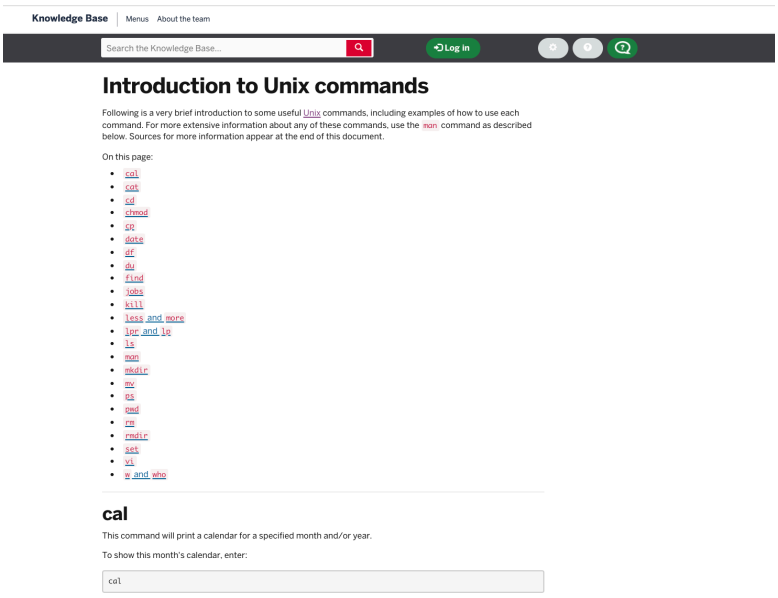
Education: BS/MS in Web Development

Goal: Present an online resource for her students to use outside of the classroom.

Stephanie is a Web Development professor who has a formal education in Web Development and many years of experience working on and teaching the subject. Her goal is to present our website as a resource to her students and to use it in class to show/demonstrate new topics. She is familiar with the material covered on our website, and can confidently refer to it. The students of her class will be referring to the website after each topic is covered in class.

# Competitive Analysis

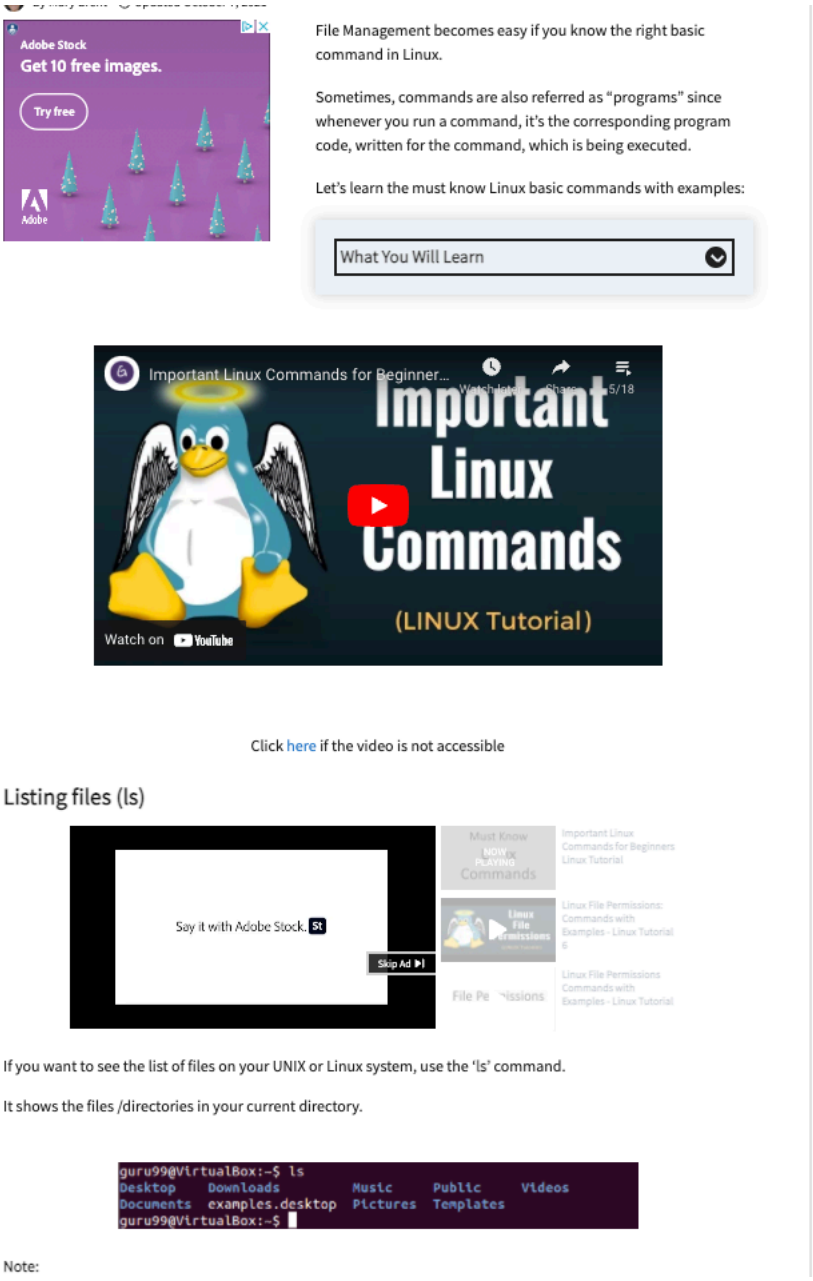
## University of Technology Services



There are a lot of things that we believe were done well on this site. The first pro on our list was this site was very well-organized.  It shows the list of commands on the page as a bulleted list of links. This is one of the first things the user sees when they visit, so that they can easily jump to the specific command that they want to learn about. These are also listed in alphabetical order to make searching for a command even easier. Another pro was that the site has command prompts that further display the particular command in use. Lastly, we liked that this site had a chat bubble button on the bottom right side of the page. When clicked, it would take you to a virtual assistant whom you could ask questions to. We borrowed this aspect of the website and made it better by incorporating the same speech bubble icon design into our own website, except we made it into a handful of hint/clue sticky notes. This way, we could guide the learning of the beginners visiting the site who may not know what questions to ask.

There were some aspects of this site that we felt were incomplete. First, there were no quizzes, so users can't test their knowledge fully. Of course, this probably isn't that big of a concern though if users just want to look at the definition of one command and then leave. Another thing the website doesn't seem to cover is how to actually get to the terminal or even a formal introduction to using Unix. This omission seems like a strange choice if their audience consists of beginners who don't even know what Unix is. For this reason, we decided to include an introduction to using the terminal and Unix on our site.

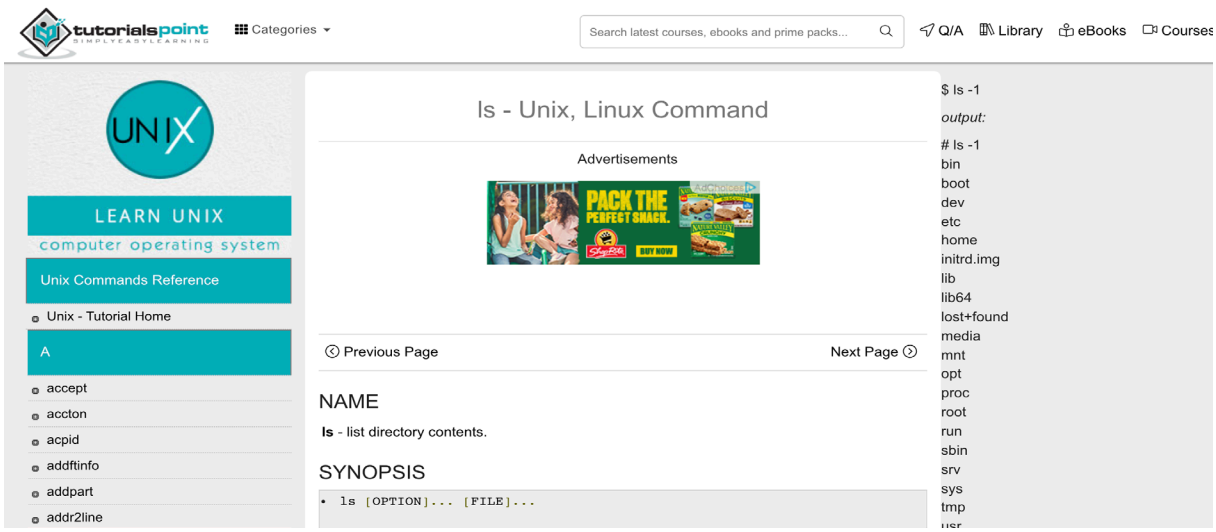
# Guru99




To start off, one pro of this site was the fact that it included a terminal screen. We decided to borrow this and make it better by also making the screen interactive as well. That way, users could type a command and actually see its behavior in front of them. Another good aspect of this site was the fact that it included helpful graphics, videos, and images to aid in learning.

One negative aspect was the fact that the color scheme seems a bit dull. Nothing really stands out to site visitors. For example, all the fonts used are the same; there are no elements that are really made distinct from each other. In this respect, the site fails to follow the “contrast” constituent of the CRAP design principles.  We aim to utilize contrast by using colors that lie opposite from each other for our site’s color scheme (as evident on our Quizzes page).

## TutorialsPoint




This site was among the best in our research. We really liked the site’s navigation. It really felt like we wouldn’t be lost from whatever link we clicked, and we always knew where we were. That’s why we decided to borrow the sidebar that this site used. We even made ours collapsible so that there is more screen estate for the site visitor. We also borrowed the site’s “previous page” and “next page” links to navigate from one command being discussed to the next. Again, these aspects really helped the web site’s navigation.

If we had to list one con from this site, it would be the fact the content seems a bit cluttered. On the right side, you can see that there appears to be a list of unknown topics or examples. Upon closer inspection, you can see that these are additional examples for the command (in the case of the screenshots above, the command is “ls”). However, it is put to the side, almost alienating it from the main body. One would think that it was unrelated or even an ad. Therefore, I think this site could try to structure itself better. Our site will be better by the simple fact that there won’t be an intimidating block of information cast to the side of our main content. We also structure our content using the “proximity” constituent of CRAP, grouping commands that are similar in behavior together (for example, “Directory Commands” or “File Commands”).

## Site Content

### Website content is grouped into three main categories:
### Getting Started
### Commands
### Quizzes

As discussed in further detail below, the Getting started section is intended for beginners.  Content in this section is likely irrelevant to users who are familiar with unix.  Because of this, we did not place individual links to this information in the navigation bar. Instead, these topics are grouped under the Getting Started tab.  There are two benefits to this;  the navigation bar is less cluttered and beginners are more likely to find this information.  A user is unlikely to click on a tab labeled “File Structure” if they do not know what a file structure is.

The commands section is given a space on the global nav.  This ensures that experienced users who need a refresher on a particular command can access this page quickly.  The commands are further organized into categories based on their function.  Because of this, new users are able to find the commands they need to use to accomplish their goal, while also providing a streamlined way for advanced users to access the information they need.  Additionally, commands are listed in the order that we recommend new users to view; starting with simple commands and ending with more advanced ones.

## Site Map

Quasar's website is mapped to allow for the efficient site and page navigation. When arriving at the website, you are met with a landing page that describes the purpose of the website. On the landing page, you are presented with options on the horizontal navigation bar which are suitable for first-time users, such as the "Getting Started" page, or options that are suitable for experienced users such as "Commands" and "Quizzes" pages.

Each page after the landing page has an expandable vertical navigation bar which can be used for page navigation to quickly view the topics on each page. The "Commands" category on the horizontal navigation bar can be used to view the commands included on our website and allows for easy access to each page. The "Quizzes" tab allows the user to quickly access the quizzes which are associated with each command topic on the website.

## Full Content

**Getting Started**

The getting started page is intended for beginners and people unfamiliar with Unix. This page is divided into three sections: An introduction to Unix, introduction to file structure, and how to access the terminal. It will also familiarize the user with the website and identify where information can be found.

**Quizzes**

There will be 5 quizzes. 4 will focus on content in the commands section below. They will ask questions about the usage of the commands and their syntax. Each of these four will be centered on one of the command categories (Text files, Directories, Navigation, and File Operations). The 5th quiz will focus on the Getting Started section and contain questions about file tree structures. An example question on a quiz might be “Of these commands, which is used to append the contents of one file to another?”

**Commands**

Below is a list of commands that we deem essential for aspiring web developers to know.

More commands may be added in the future for both convenience and to broaden the audience of the site.

**Text Files**

vi: launches a text editor called Vi.

Options

-R: opens the file in read-only mode

+: positions the cursor to the last line of the file

+n: positions the cursor the the n line of the file, where n is a number

> **Syntax: vi +10 quasar**
>
> Opens the file “quasar” and positions the cursor to the 10th line of the file

cat: reads a file and sends the contents to an output. Can be used to append files or output the contents into the terminal window.

**Options**

-b: starting at 1, number non-blank output lines

-n: starting at 1, number all output lines

-u: Output is displayed unbuffered

**Operators**

> redirects the output to the specified location, rather than the standard output. If the destination file does not exist, it will be created. If it does exist, it will be overwritten.

>> appends the end of an existing file with the output. If the file does not exist, it will be created.

**Syntax: cat [option] [files]**

Example 1: cat quasar

The contents of quasar are printed in the terminal

Example 2: cat file1 file2 > file3

Takes the contents of file1 and file2 and concatenates them into file3

**Directories**

mkdir: creates a directory or directories with the specified names

Options

-p: if the parent directories do not exist, create them

**Syntax: mkdir [option] [directory]**

[directory] can be a name or a filepath

**Example: mkdir -p quasar/example**

Creates a directory “quasar” with child directory “example”

rmdir: removes the directory with specified name. The directory must be empty. To remove a directory with files, see the rm command.

**Syntax: rmdir [directory]**

Example: rmdir quasar

Removes directory named “quasar”

**Navigation**

ls: Lists the files in the current directory

Options

-l: lists in long format, which contains information about the files’ size, ownership, permissions, and the last time they were modified.

-a lists all files, including ones that would not be listed otherwise.

**Syntax: ls [option]**

Example: ls -a

Lists all files in the current directory

pwd: displays the full pathname to current directory

Syntax: pwd

cd: changes the current working directory to the specified directory.

Syntax: cd [directory]

[directory] can be the absolute path or the relative path to find the directory.

Example 1: cd /example

changes working directory to directory named “example”

Example 2: cd ..

changes working directory to the parent directory

**File operations**

cp: Creates a copy of the specified file.

Syntax: cp file1 file2

Example: cp example  example2

Copies example.text, names the copy example2

rm: Removes the specified file. Can be used to delete directories using options. Deleted items are not  recoverable.

Options

-i: asks for confirmation before deleting the file

-r: deletes the specified file. If a directory is specified, deletes all files and subdirectories (use with caution).

-f: overrides confirmation when deleting write-protected files.

**Syntax: rm [option] filename**

Example: rm -r Stuff

Removes the directory named “Stuff” along with all of its contents mv: Moves the file to the specified location. Can also be used to rename a file if the destination is a

Options

-i: asks for permission before overwriting a file

-f: forces the move

**Syntax: mv [option] filename destination**

Example 1: mv -f example  /usr

Forces the move of example to directory “usr”

Example 2: mv oldfile  newfile

Renames “oldfile ” to “newfile ”

chmod: Changes the permissions of a file or directory

Modes

Who:

u = user

g = group

o = other

a = all (default, if not specified)

**Opcode:**

\+  add permission

-  remove permission

=  set permission

Permission:

r = read

w = write

x = execute

**Syntax: chmod [mode] filename**

Example 1: chmod rwx example

Sets file permissions to read, write, execute for all

Example 2:  chmod u=rwx g=-r o=rw

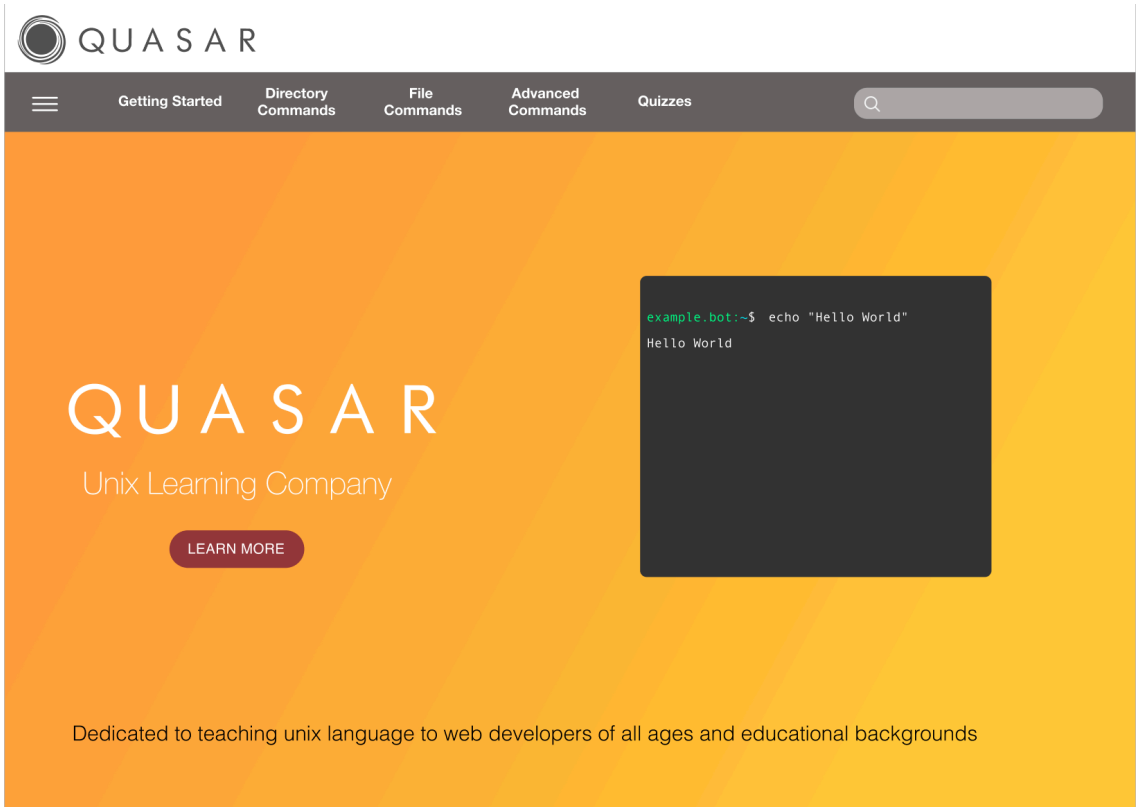
Sets user permissions to read, write, execute.

Removes read permission from group.

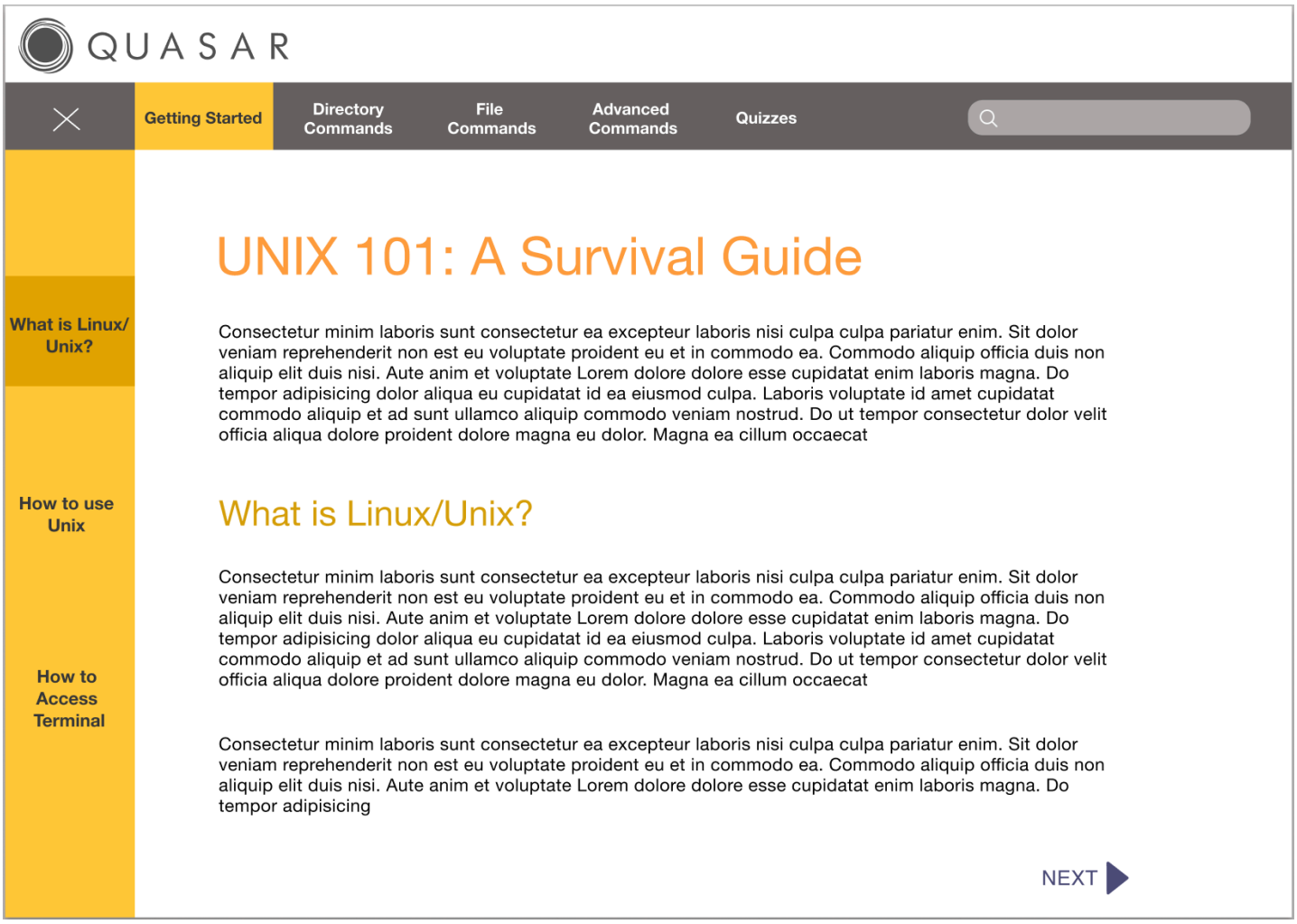
Sets other permissions to read, write.

## Design

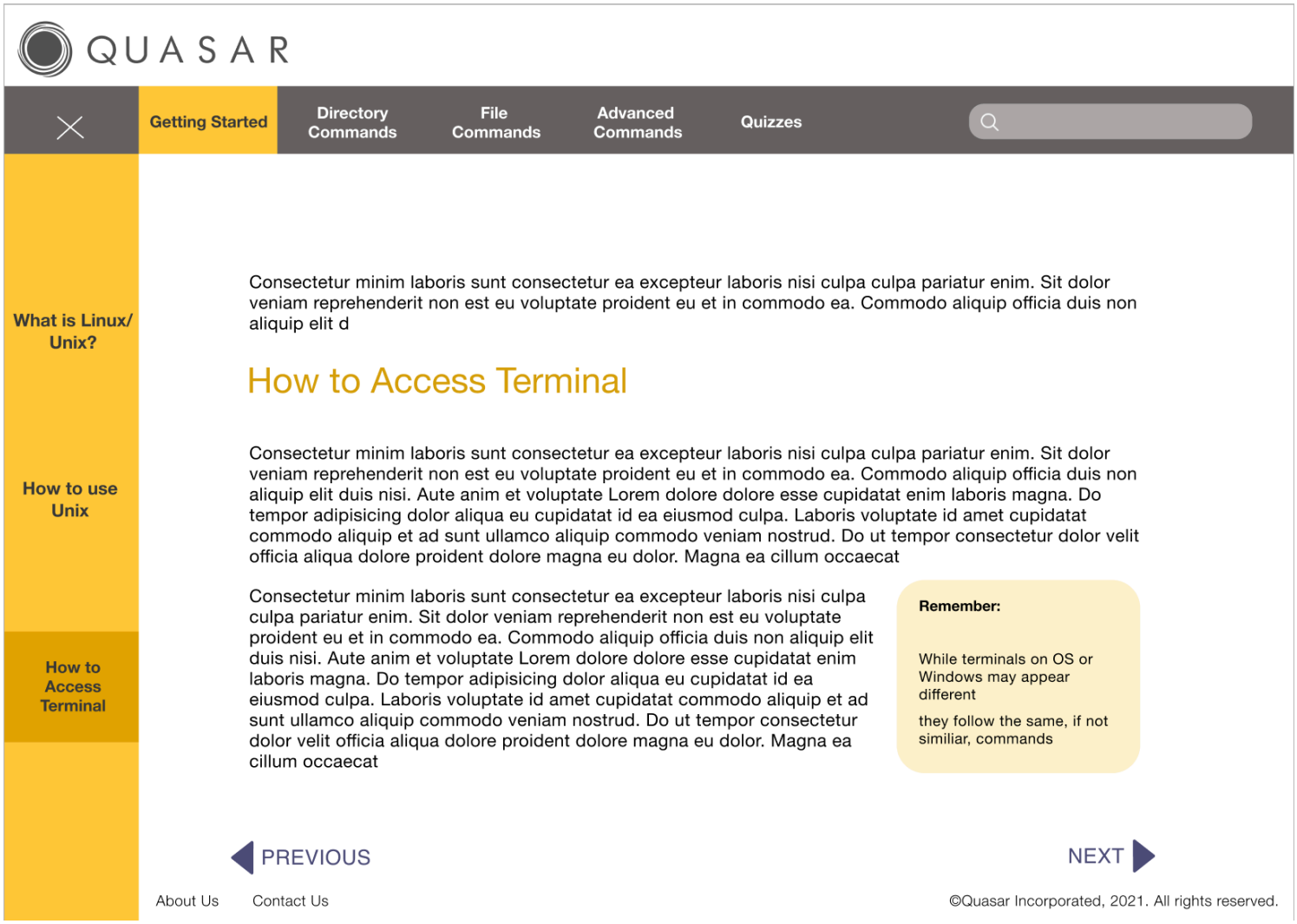
### Website Sketches



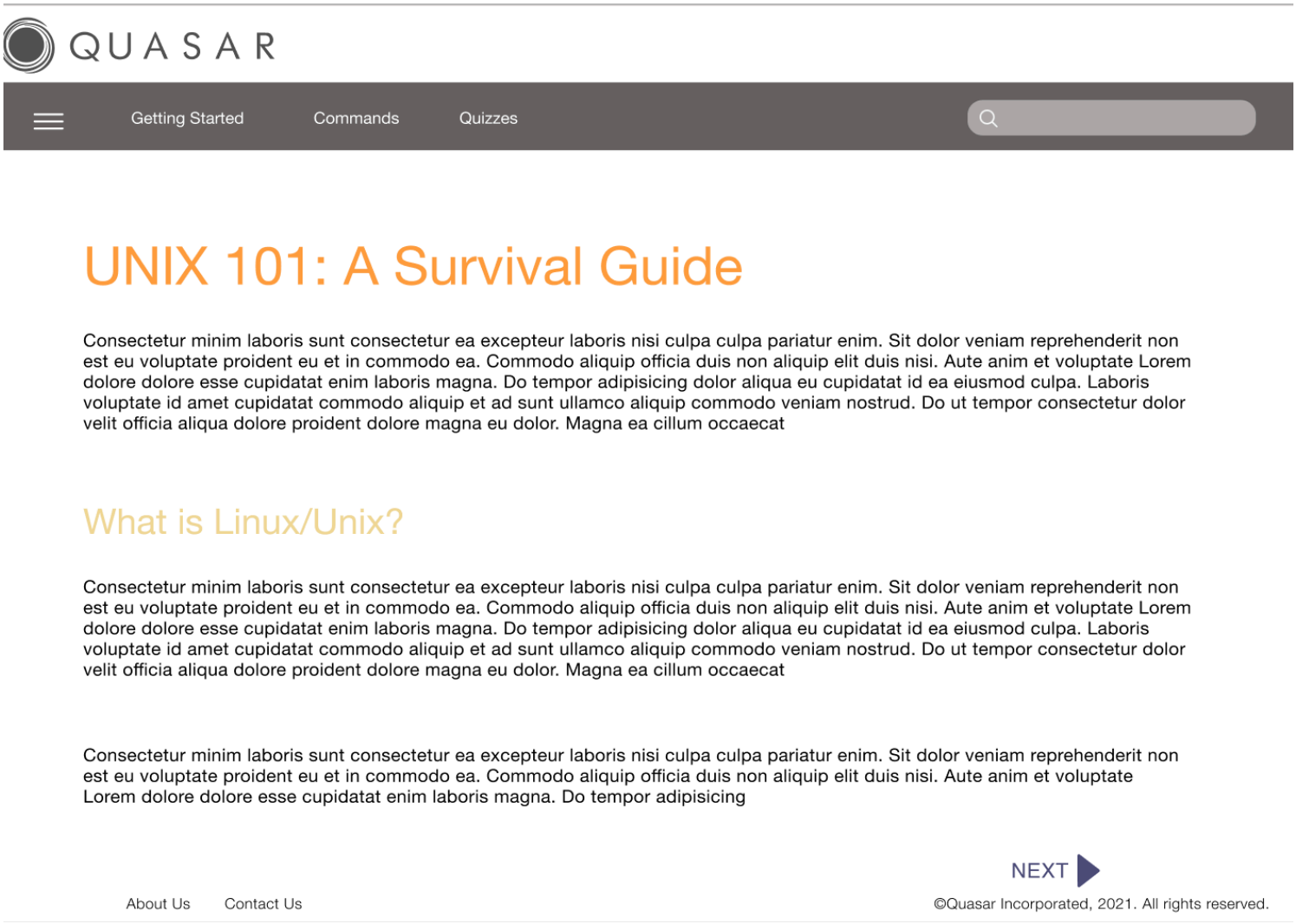

Landing page—introduces the user to the site and prepares introductory navigation



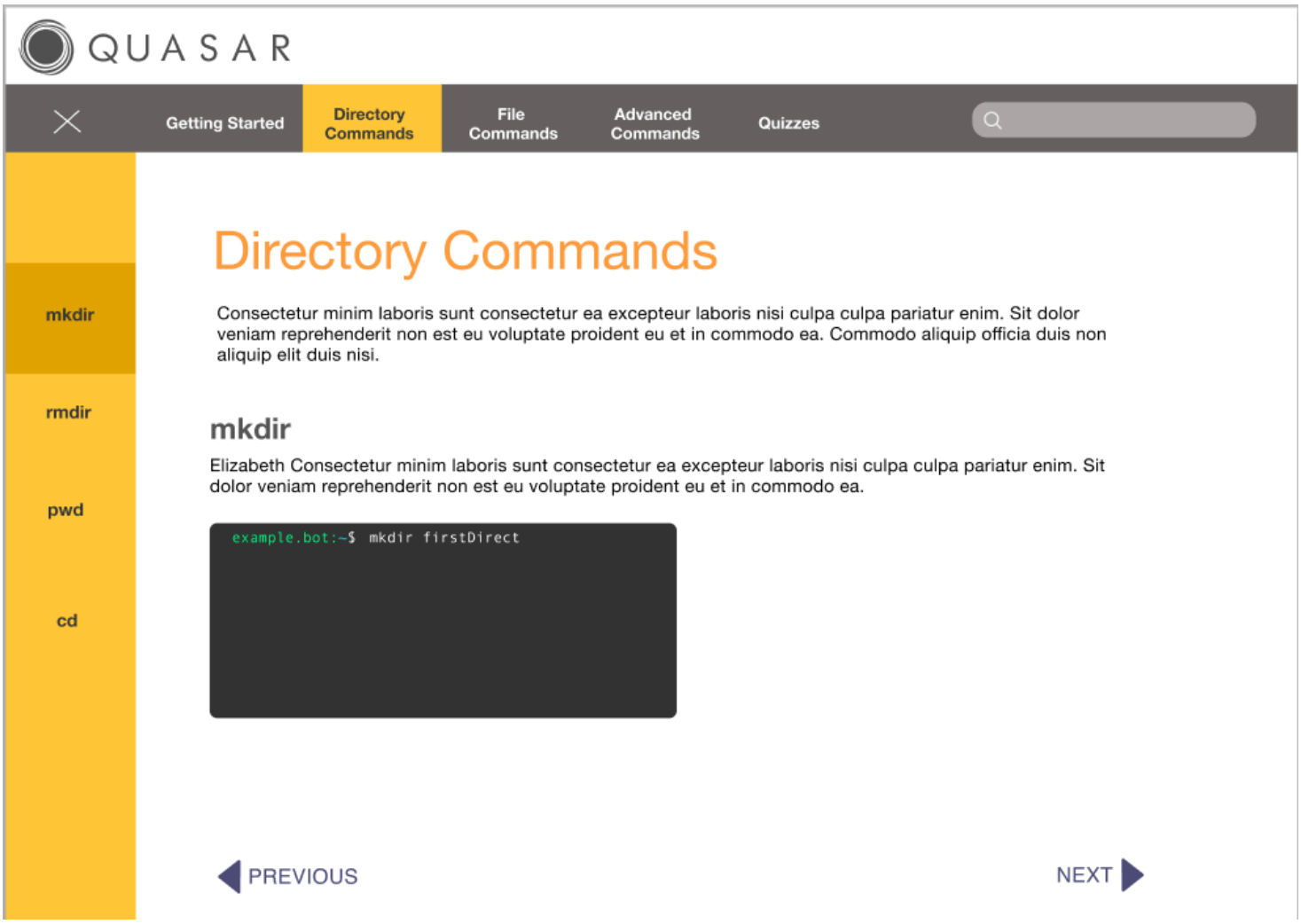

What is Linux/Unix? under Getting Started—–defines what Unix is, the difference between Linux and Unix, and how to access terminal based on the device of the user. The page contains all of the content in one but the next button allows the user to go to the next topic.




How to Access Terminal—has a remember feature that reminds the user of the most important issue in a section, and allows the user to remember the most important information.

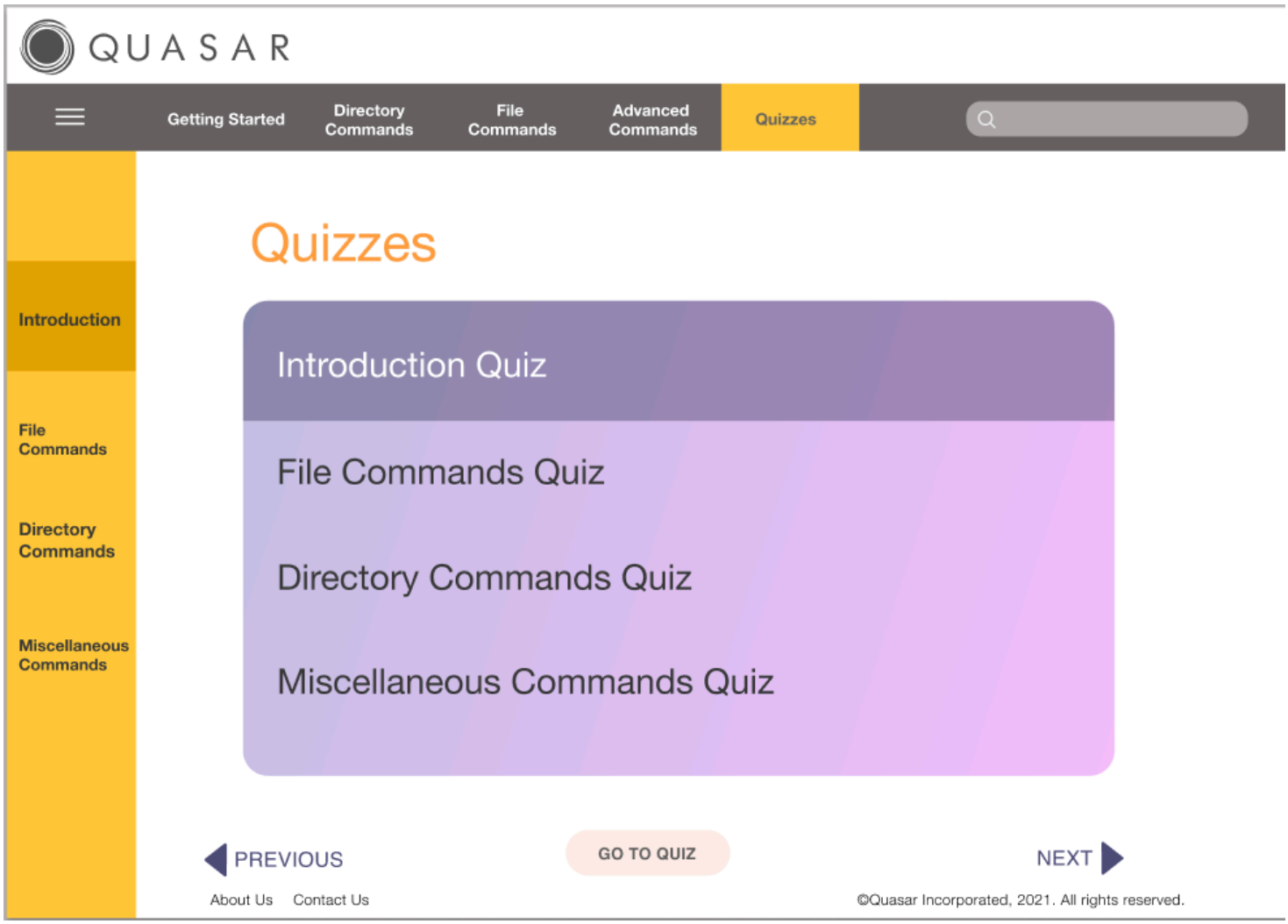



UNIX 101: A Surivival Guide part 2—Side nav can disappear at the user's will due to the hamburger menu.

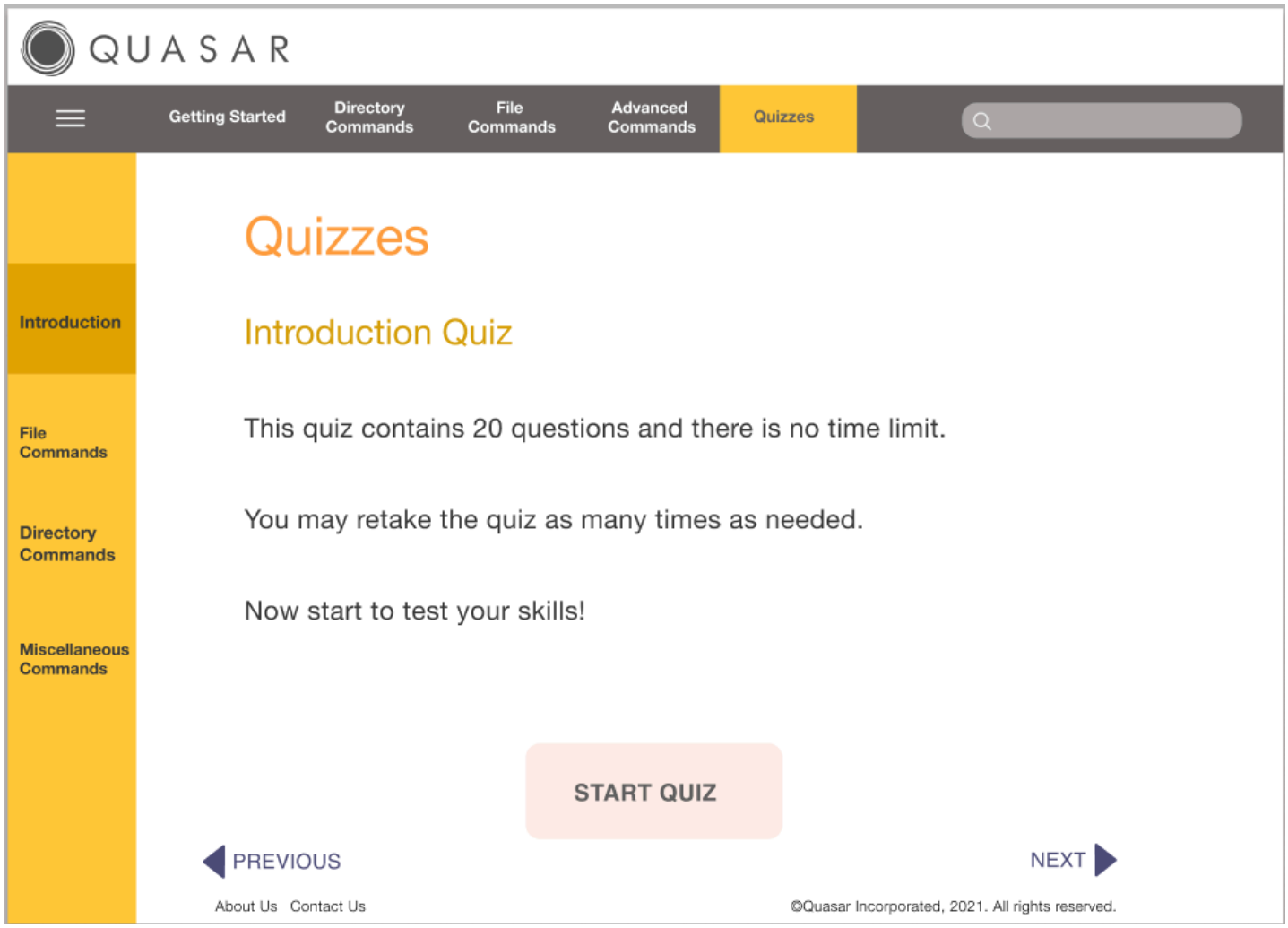



Commands section—the main content, contains a terminal that allows the user to test what they are learning.

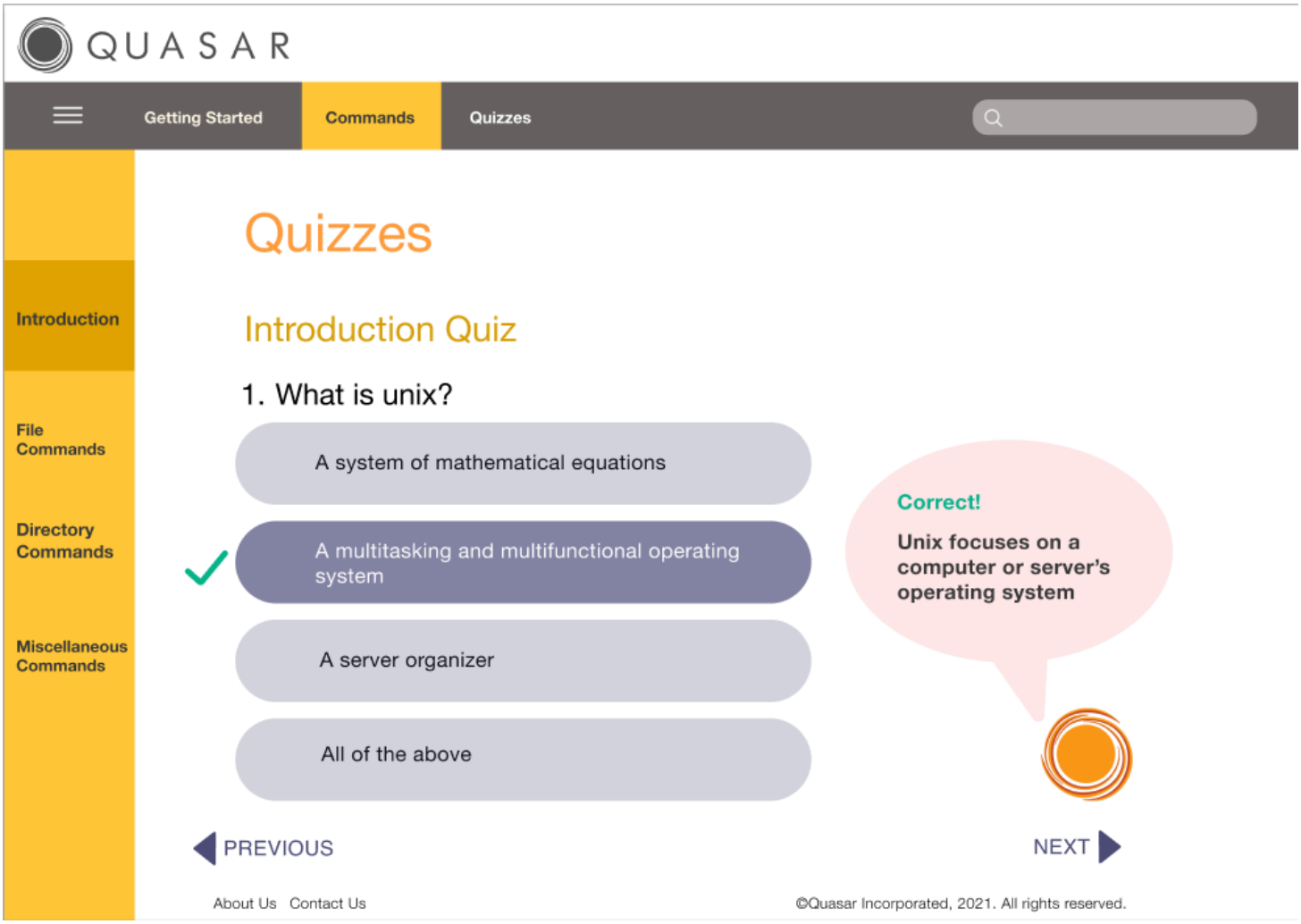

Dropdown menu—when the user hovers over the top nav, a more complex nav appears.




Quizzes—allows the user to first click what quiz they wish to take.

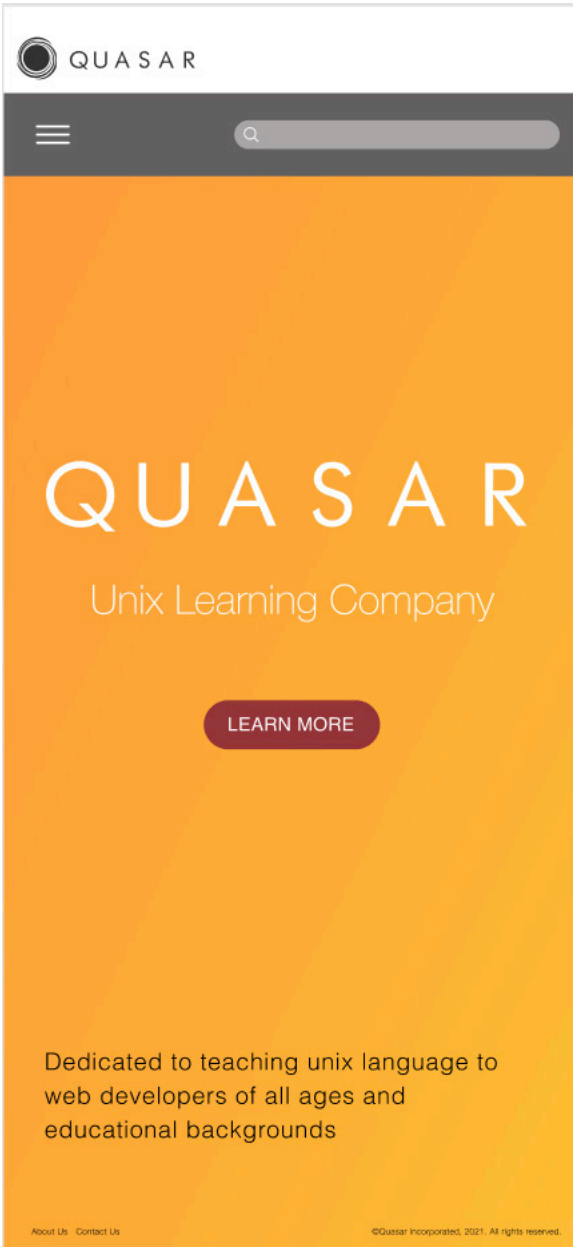



Quiz starting page——introduces the rules of a quiz

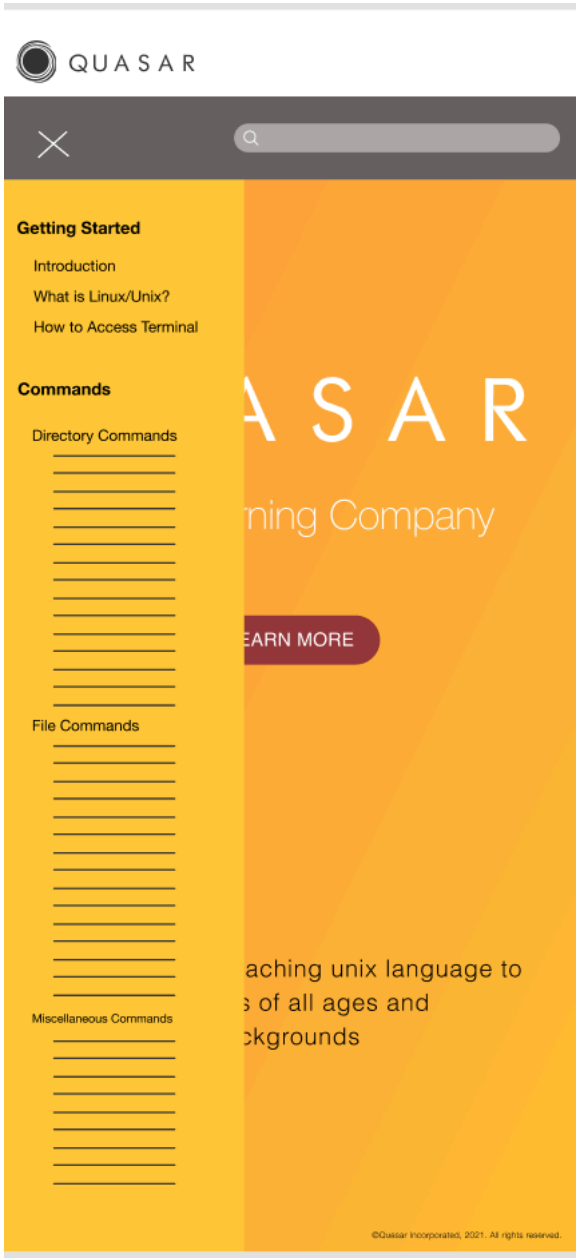



Quiz—shows if the answer is correct or wrong and the mini quasar explains the reasoning behind the answer.

# Mobile Sketches



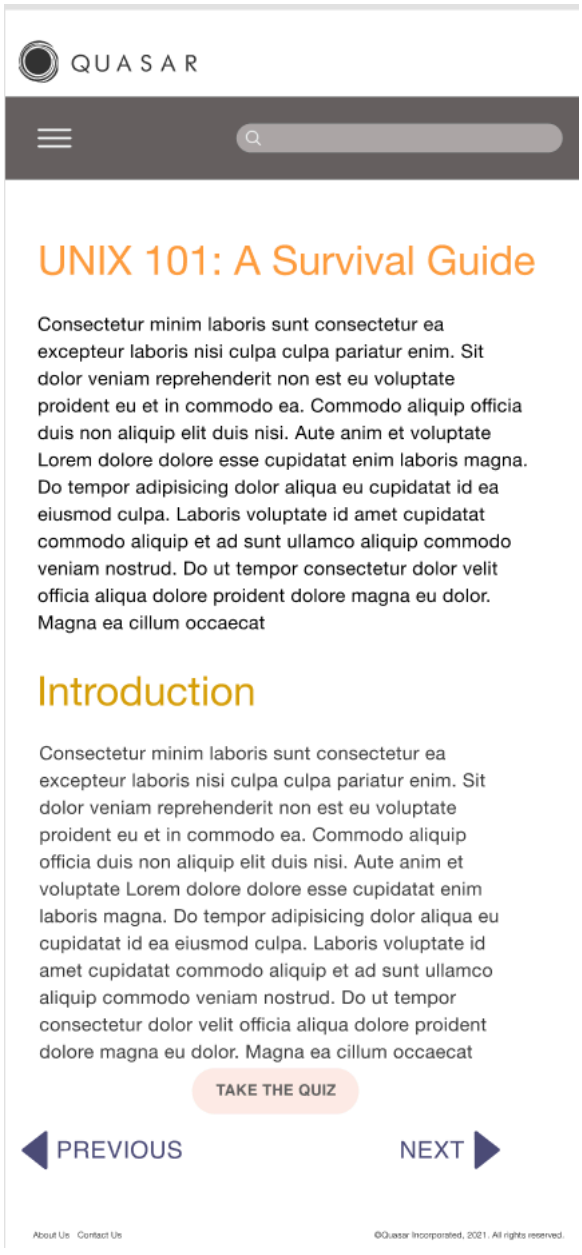

A




B

A: Landing page—introduces the user to the site.

B: Getting started page—nav bar is completely hidden and buttons are bigger for mobile



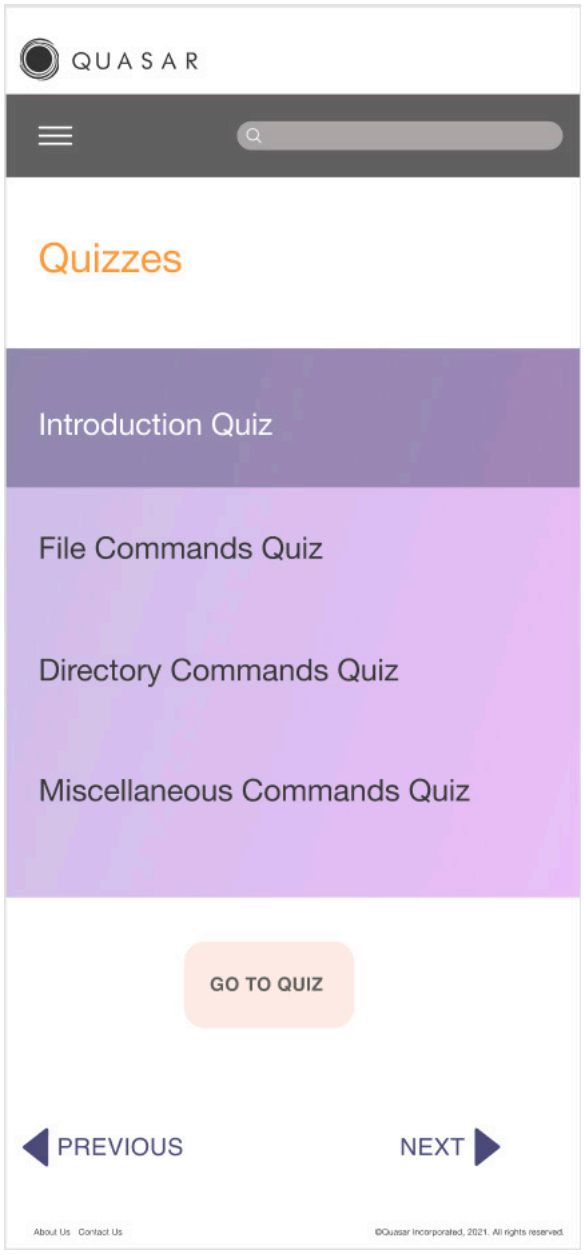


**A**

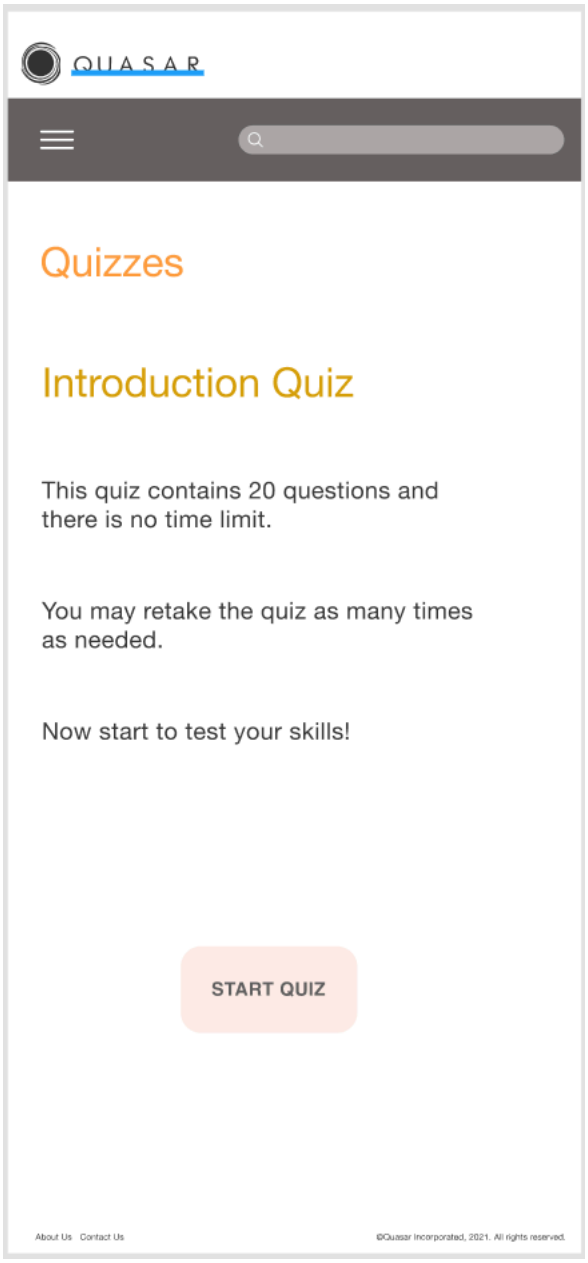
**B**

A; Hamburger menu—drops down to show full nav

B: Commands—Mobile has more space and the terminal takes up more of the page.



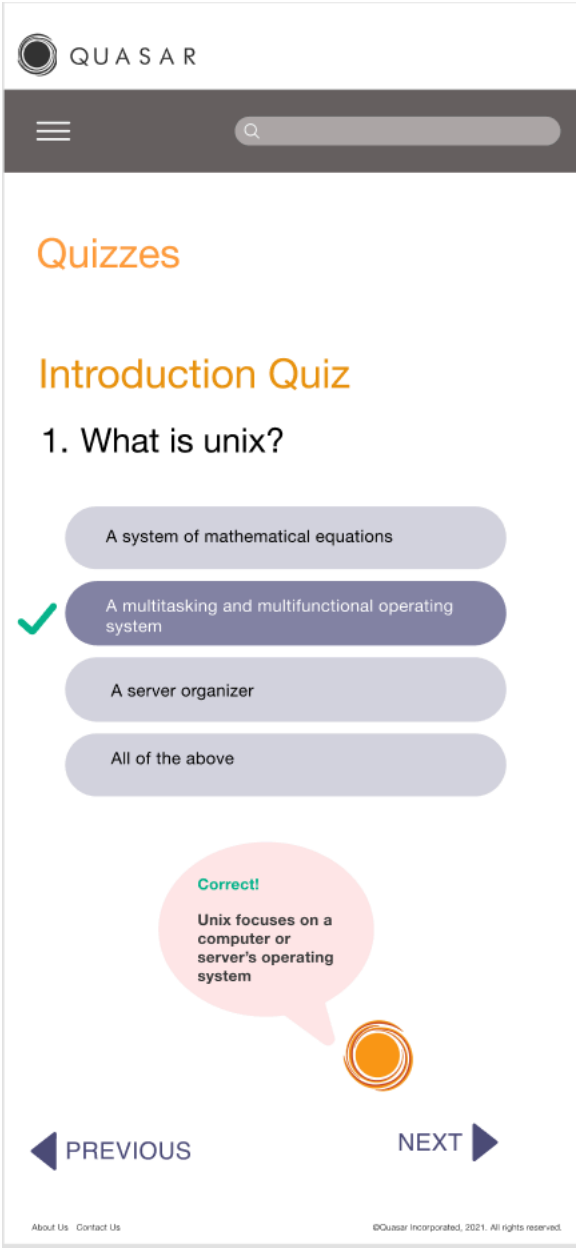

A




B

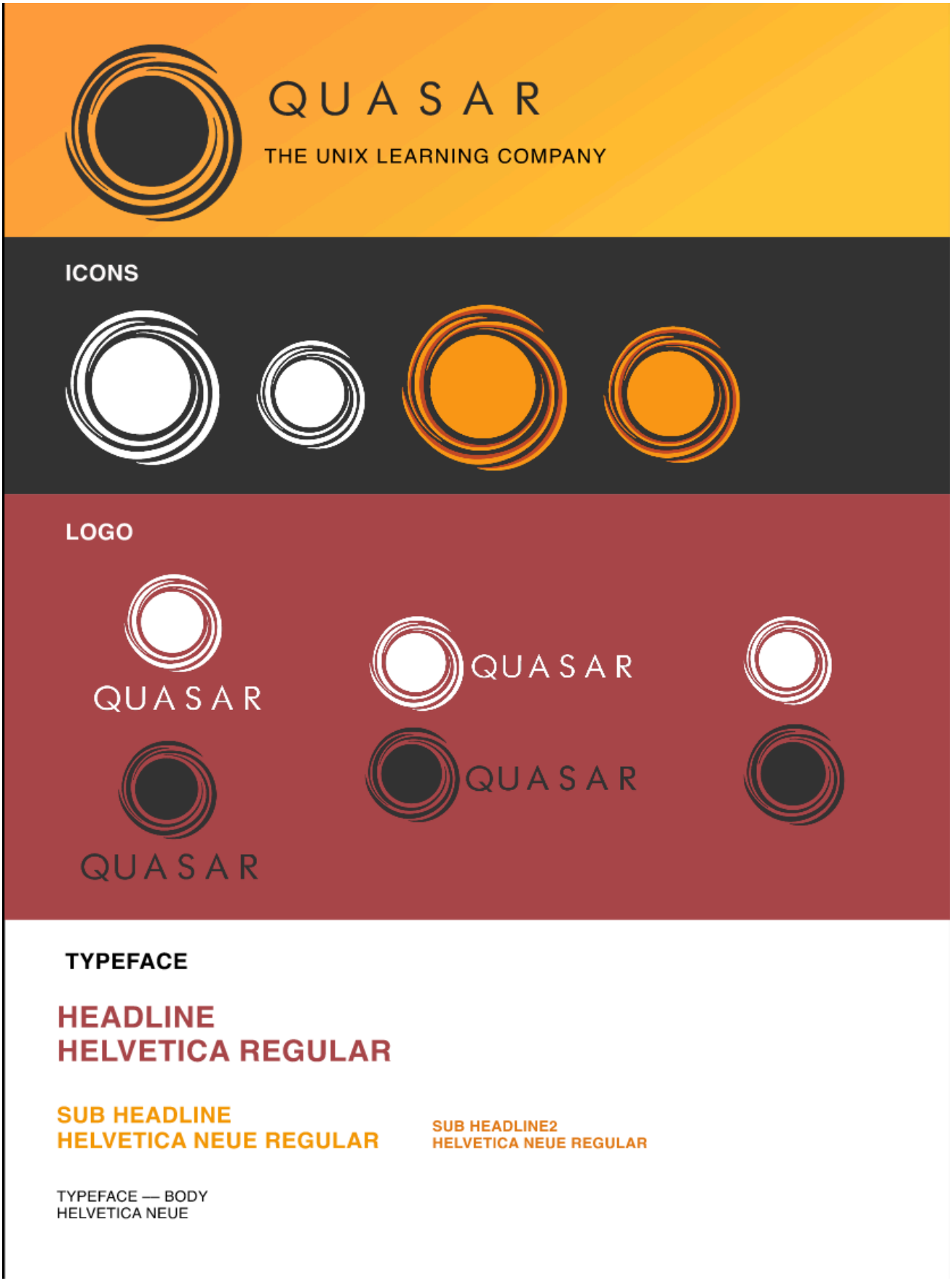
A: Quiz section—mobile takes up more space and choices are bigger so the user has an easier time clicking them

B: Introduction page for Quiz




Quiz answer page—same concept as the website but bigger for comprehension

# Style Guide


QUASAR
THE UNIX LEARNING COMPANY
ICONS
LOGO
QUASAR
QUASAR
QUASAR
QUASAR
QUASAR
TYPEFACE
HEADLINE
HELVETICA REGULAR
SUB HEADLINE
HELVETICA NEUE REGULAR
SUB HEADLINE2
HELVETICA NEUE REGULAR
TYPEFACE — BODY
HELVETICA NEUE

## COLOR PALETTE

ACCENT COLOR

ACCENT COLOR

BACKGROUND COLOR

TEXT COLOR

ACCENT COLOR

## BUTTONS

CLICK FOR MORE

NEXT

TAKE THE QUIZ

LEARN MORE

HOVER:

CLICK FOR MORE

NEXT

TAKE THE QUIZ

LEARN MORE

## HINTS

REMEMBER:

.....

HINT:

,,,,,,

# Requirements

## Essential Requirements

The site has five (5) essential requirements.

One, to be accessible to different educational groups.
Whether the user is a professor with a complete understanding of Unix or is a first-year student who has no understanding of Unix or coding techniques.

Two, to be accessible on all platforms.
The site should operate, or fail gracefully but still be understood, on all browsers. It should operate smoothly on any device, so it may be accessible in any location.

Three, the site should be understood by the common person.
It will not contain any jargon that makes understanding Unix incomprehensible. Any use of jargon should be thoroughly explained, especially each time it is used, to reduce confusion.

Four, the quizzes on the site should be aimed to only increase a student's understanding. Wrong answers should be explained and correct answers should also be given a detailed understanding. Quizzes should be a learning opportunity.

Five, there must be no assumptions when constructing content.
The content, quizzes, and navigation of the site must be comprehensible to any group. Anyone should be able to navigate the site, understand its content and operations.

## Desirable Requirements

It is desired for the site to be as aesthetically pleasing as possible. While functionality and comprehension are the highest priority, making the site pleasing to the user is a desirable goal.

It is also desirable for the site to be used by a student with a complex understanding of Unix or by a professional web developer looking for a refresher of definitions. The site should be simplistic in definitions but not appear to be demeaning to those who have prior knowledge. It should also provide a gateway to more complex content.

What makes these aspects desirable and not essential is that this caters to a subcategory of users. The focus of the site should be for educational purposes towards students with no prior knowledge. Knowledgeable students or professionals are not the main audience but are considered.

## Global/Accessibility Considerations

The site should be accessible on a global scale and for disabled students. Globally, the site should contain language that is easy to translate and should be understood by students who may not have English as their first language. The site should appeal to those on a global level since it is not focused on content about any one country, it should be understandable for anyone from a different country than the United States. The site will also contain a lack of American cultural references so it may appeal to others on a global scale.

In terms of accessibility, the site will be compatible with text-to-speech reading. It should also be readable by anyone with a colorblind issue. Finally, the site will also contain language that is simple enough to be understood by anyone despite any intellectual stance and its language will not contain anything that may dismay or discourage the user.

## Deployment Environment

The deployment environment will be LAMP, which is free and open-source software. It is most suited for Linux language and contains MySQL, apache, and PHP language. It will suit the site both in its functionality, as it relies on PHP language and SQL databases, and in terms of content, as LAMP focuses on Linux which is one of the main aspects of a site that teaches Unix language.

## Conclusion

With Quasar, we will provide users of any skill level with easy-to-understand information as well as a comprehensive website that makes it easy for our users to find what they are looking for. We aim to make our website a productive learning space by giving users access to the command lines where users can see how commands work right on the site. Our mobile site will also ensure that users can work and study on the go with a user-friendly and well-designed mobile version.

To satisfy the user, we first considered their perspective. The website is curated for future web developers learning about Unix. Therefore, the site contains simplistic language and does not assume the user has any preexisting knowledge of Unix. The site is also made to continuously test the user, for them to improve their skills. Quizzes have no time limit so the user does not feel stressed and the correct answers are explained to reaffirm their beliefs. Terminals are implemented into the site so the user may test their skills in a controlled environment. When considering the other categories of users—teachers and professionals—the site has direct definitions that appear first, then embellished explanations afterward. This is to allow professionals and teachers to obtain the definition they require first (as they are more likely to scan a page trying to find what they need) and still provide a more in-depth answer later.

## Appendix

1. General Adversarial Network (GAN)

    Generates faces for personas