

Prototype Screenshots for Downloader Application

Captain CyBeard: Neil Before Us

Ryan Breitenfeldt | Noah Farris
Trevor Surface | Kyle Thomas

April 29, 2020



Washington State University Tri-Cities
CptS 423 Software Design Project 2

List of Figures

1	First page, ready for a url to a cloud platform.	1
2	URL entered that goes to cloud account.	2
3	Username to cloud platform entered.	2
4	Password to cloud platform entered. Note: Javascript doesn't allow for a popup with two text boxes in the same window or two windows at the same time. The actual implementation will include both the username and password entry boxes in the same popup window.	3
5	Files available for downloading in the root directory of the cloud storage.	4
6	All files selected for downloading.	4
7	Download button pressed, browser prompts where to save files.	5

1 Introduction

The following figures are screenshots for the *Downloader* Prototype. The prototype was created with HTML, CSS and Javascript and opened with a web browser (a web server was not used to serve the files). The *Downloader* application is an application where users will enter a URL that goes to their external cloud service and then the application will have the users authenticate to their external cloud service. After authentication, the user can navigate their directories on the cloud platform and select which files to download. The screenshots show this process visually.

Figures 3 and 4 must be two separate steps due to the limitations of vanilla Javascript, however when the application is actually implemented with Django these two inputs from the user will be in the same window.

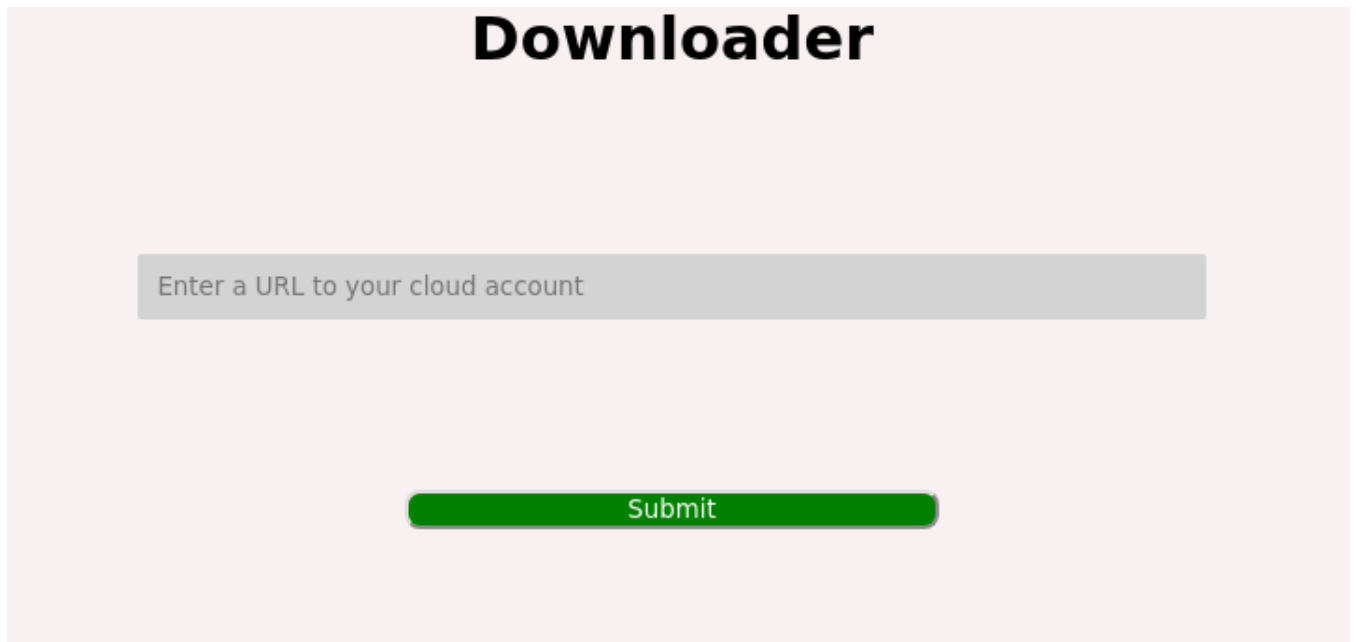
The image shows a web page with a light pink background. At the top, the word "Downloader" is written in a large, bold, black sans-serif font. Below this, there is a light gray rectangular input field containing the placeholder text "Enter a URL to your cloud account". Centered below the input field is a green rectangular button with rounded corners and a thin black border, containing the word "Submit" in white text.

Figure 1: First page, ready for a url to a cloud platform.

Downloader

mycloud.aws.com

Submit

Figure 2: URL entered that goes to cloud account.

This page says

Username:

uname

Cancel OK

mycloud.aws.com

Submit

Figure 3: Username to cloud platform entered.

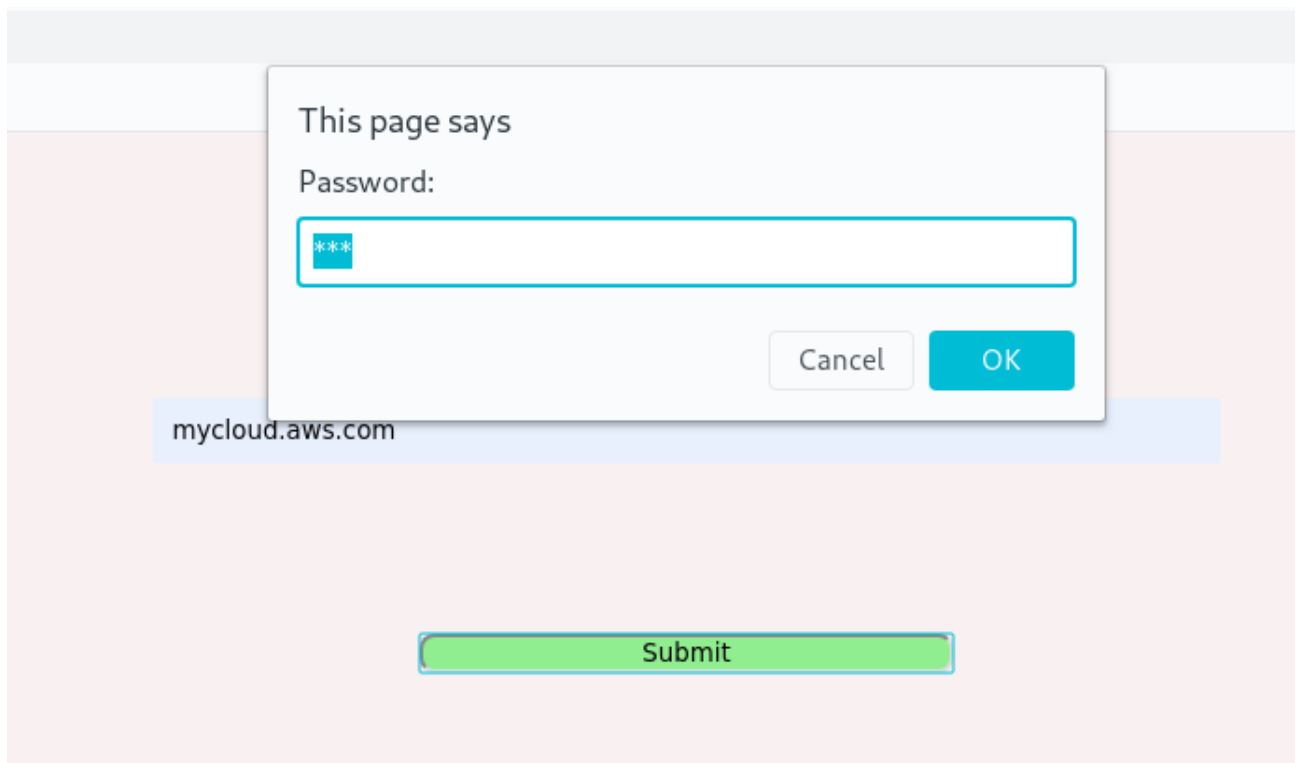


Figure 4: Password to cloud platform entered. Note: Javascript doesn't allow for a popup with two text boxes in the same window or two windows at the same time. The actual implementation will include both the username and password entry boxes in the same popup window.

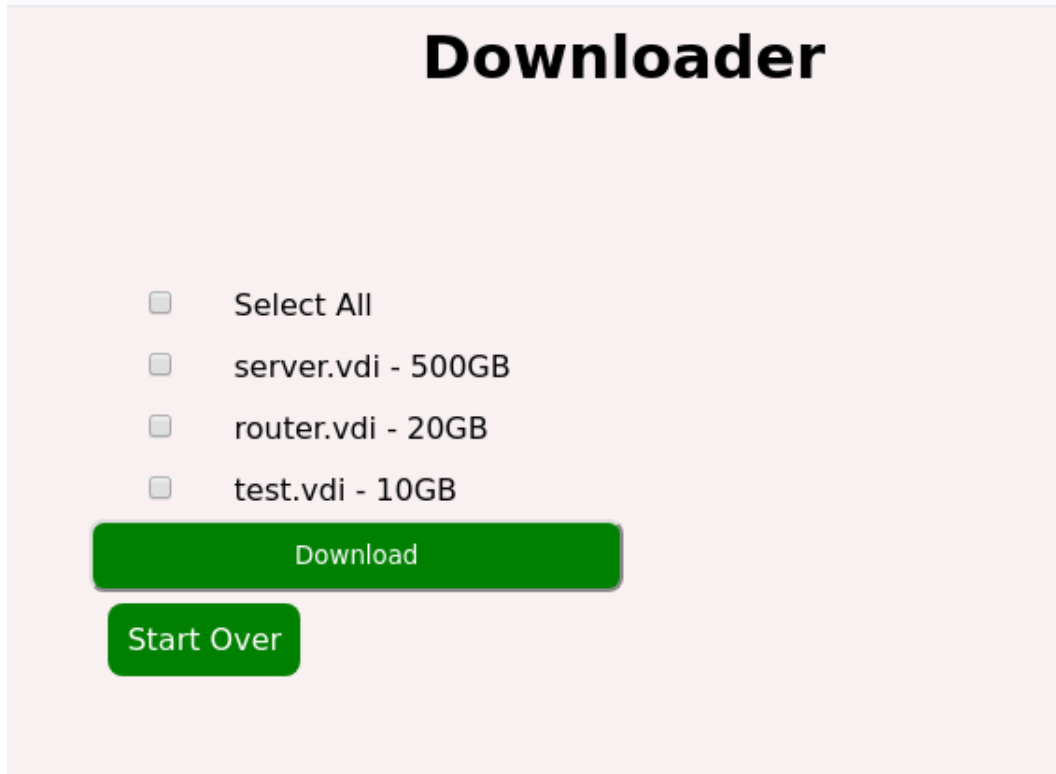


Figure 5: Files available for downloading in the root directory of the cloud storage.

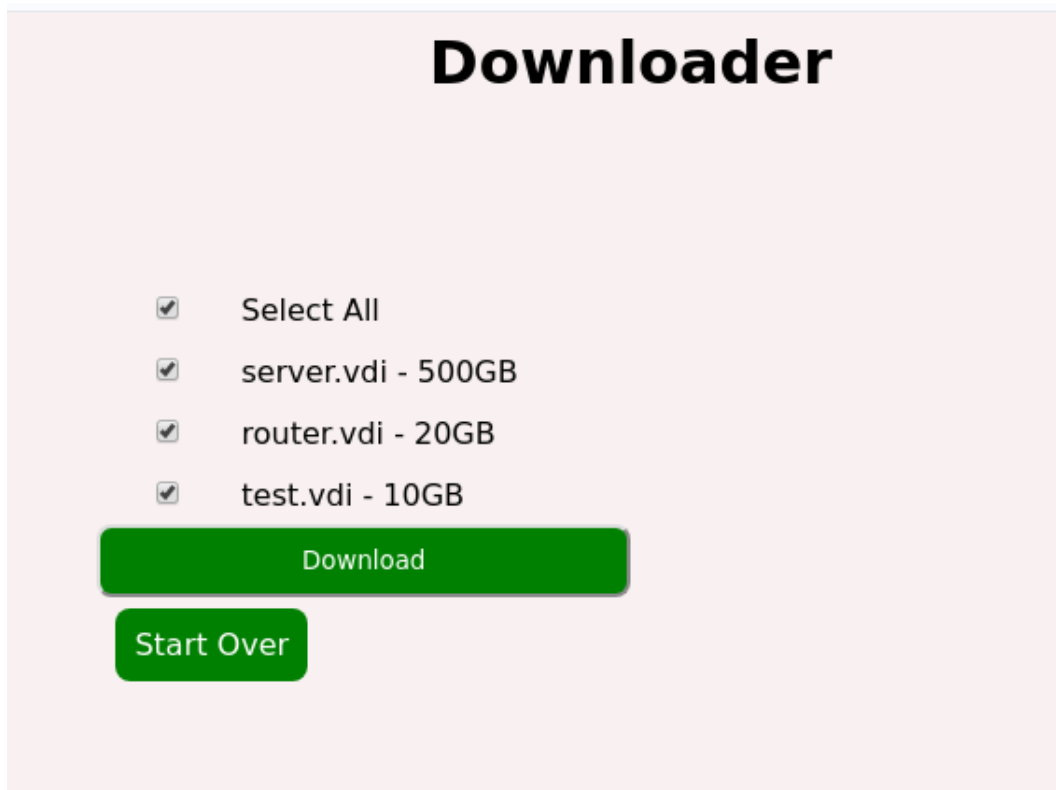


Figure 6: All files selected for downloading.

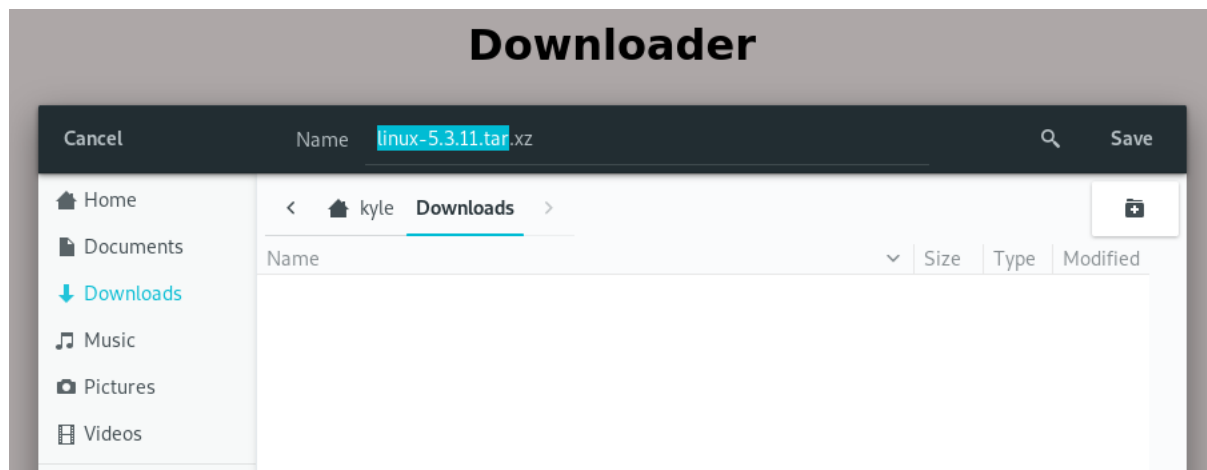


Figure 7: Download button pressed, browser prompts where to save files.