Logo%20Main%20200

**Boston University**

**Electrical & Computer Engineering**

**EC464 Capstone Senior Design Project**

User Manual

Coastline Prediction using Existing Climate Change Models

Submitted to

MathWorks

1 Apple Hill Drive

Natick, MA 01760

508-647-7001

service@mathworks.com

by

Team 18

Team Sea Rise

Team Members

Stacia Kolodziejski [skolodz@bu.edu](mailto:email1@bu.edu)

Wenyu (Jessica) Hu [wjhu@bu.edu](mailto:email1@bu.edu)

Saif Alblooshi [sjalbloo@bu.edu](mailto:email1@bu.edu)

Submitted: April 17, 2023

#### User Manual

#### Table of Contents

[User Manual 1](#_heading=)

[Table of Contents 1](#_heading=)

[**Executive Summary (Stacia Kolodziejski) 2**](#_heading=h.30j0zll)

[**1 Introduction (Stacia Kolodziejski) 3**](#_heading=h.1fob9te)

[**2 System Overview and Installation (Stacia Kolodziejski) 4**](#_heading=h.3znysh7)

[2.1 Overview block diagram 4](#_heading=h.2et92p0)

[2.2 User Interface 4](#_heading=h.tyjcwt)

[2.3 Installation, setup, and support 6](#_heading=h.3286smoniq1x)

[**3 Operation of the Project (Stacia Kolodziejski) 7**](#_heading=h.4d34og8)

[3.1 Operating Mode 1: Normal Operation 7](#_heading=h.2s8eyo1)

[3.2 Operating Mode 2: Abnormal Operations 7](#_heading=h.17dp8vu)

[**4 Technical Background (Wenyu (Jessica) Hu) 8**](#_heading=h.26in1rg)

[4.1 Elevation Data 8](#_heading=h.mm7rdb78l2bc)

[4.2 Prediction Data 9](#_heading=h.2p5r5ve0v54w)

[4.3 Final Function - Analysis 9](#_heading=h.2gcfskzgsp47)

[**5 Relevant Engineering Standards (Stacia Kolodziejski) 11**](#_heading=h.lnxbz9)

[**6 Cost Breakdown (Stacia Kolodziejski) 12**](#_heading=h.35nkun2)

[**7 Appendices (Stacia Kolodziejski) 13**](#_heading=h.1ksv4uv)

[7.1 Appendix A - Specifications 13](#_heading=h.44sinio)

[7.2 Appendix B – Team Information 13](#_heading=h.2jxsxqh)

[7.3 Appendix C – Technical References 13](#_heading=h.tm2h9oby81xz)

(Right click on Table of Contents to update fields and page numbers automatically)

# 

# 

# 

# 

# 

# 

# 

# 

# Executive Summary (Stacia Kolodziejski)

Team Sea Rise

18

Global warming and melting glaciers are causing sea levels to rise which exposes various coastal communities around the world to risks of erosion and flooding. These communities would benefit greatly if they knew in advance how and when their homes will be affected. To give these communities the information they need to prepare and adapt to the consequences of sea level rise, the Coastline Predictor visualizes the coastlines up to 100 years in the future.

Our product, Coastline Predictor, developed with MatLab. It will use current elevation data in addition to sea level prediction data, analyze this data, and then generate a predicted coastline. The Coastline Predictor displays new coastlines based on future time instead of water level rise depth, so that users do not need to look for predicted water levels themselves. We hope that our product can help coastal communities plan in advance, for what will happen to their homes.

# Introduction (Stacia Kolodziejski)

As the world emits more carbon emissions into the atmosphere, the consequences of climate change will start to worsen. Humans have already begun to see the consequences of climate change worldwide, like melting glaciers and rising sea levels. This consequence in particular, directly affects communities which live on the coasts around the world. These coastal communities will start to experience flooding, coastal erosion, and even some land degradation as well as a consequence of rising sea levels. In order for these communities to adapt and prepare for these conditions, they will need to know how much their homes will be affected and when.

The effects of climate change are still being researched and predicted as the world’s carbon emissions worsen from year to year, but the effects of climate change communities are already experiencing today need to be addressed. These affected communities need to be able to adapt accordingly for any future flooding. The Coastline Predictor can give coastal communities across the United States a better understanding of when they will be affected and by how much, if any.

The Coastline Predictor will give these communities an accurate and effective resource to help prepare and adapt to the consequences of rising sea levels from year to year. By using the Coastline Predictor, communities can see when their homes will be flooded depending on how much sea level rise they will experience every 10 years into the future. This function works for any region of the United States, Canada, and Mexico and shows the changing coastline every 10 years up to the year 2150.

The Coastline Predictor can accurately predict and visualize the coastline up to 10 meters of resolution and will present a comparison to what the coastline of the selected region looks like in the present day versus the future coastline of the year chosen by the user. The user will be able to see the difference of the two coastlines from a color coded, detailed map of the specific region.

When using the Coastline Predictor, the function has to process two large datasets which can take up a lot of memory on the harddrive of the user’s device. When the function runs too many times in a row, MatLab may crash due to an overdrive of data processing and not enough memory on the device’s harddrive. This poses a few problems for the user, however, the way to fix this issue is by reopening MatLab and rerunning the function. Other user interface and user issues that may come up will be discussed later in this manual.

# System Overview and Installation (Stacia Kolodziejski)

The Coastline Predictor uses MatLab software, elevation data from the United States Geological Survey (USGS), and predicted sea level rise data from the National Oceanic and Atmospheric Administration (NOAA) to create a visualization of a new coastline for any city chosen. This new coastline will show the user what the predicted amount of sea level rise will do to the present existing coastline in terms of flooding and coastal erosion.

## Overview block diagram



*Figure 2.1. System Overview of the Coastline Predictor.*

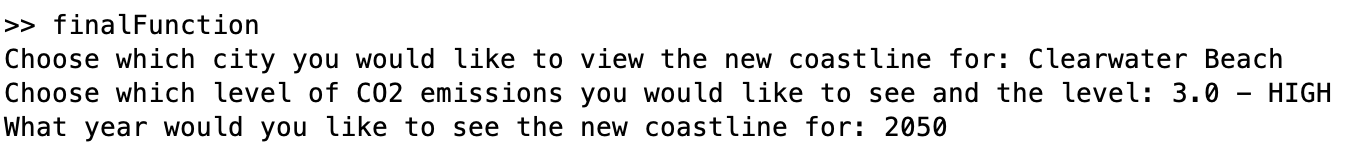
## User Interface

The user interface includes three initial prompts, where the user can input which city, which ‘scenario’, and which year they would like to view the new coastline for. The user will be presented with a list of cities that are not able to be chosen, but other than those cities, the Coastline Predictor works for every part of the U.S.

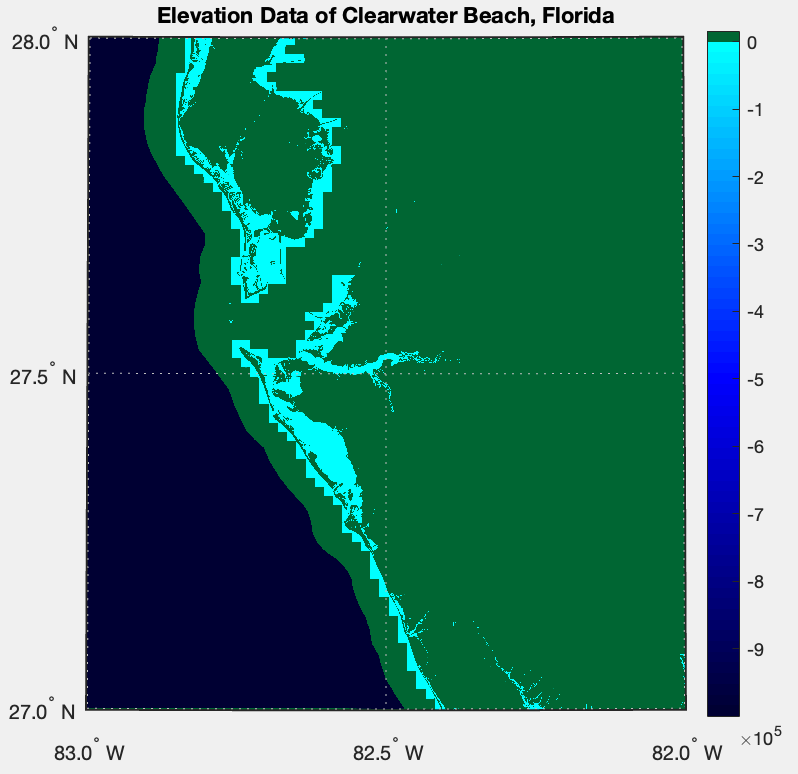
The ‘scenario’ prompt for the user to answer is based on high, medium, and low levels of carbon emissions depending on the amount of sea rise. The 0.3, 0.5, 1.0, 1.5, and 2.0 are depths, in meters, that the sea will rise depending on the levels of carbon emissions by the year 2100. The user will be prompted to input their preference indicating which depth they want to choose with a space and a dash with another space and type in all capital letters the level (high, medium, low) they would like to see as well. Examples of each prompt and answer from the user are shown below in Figure 2.3.

Once the user enters their preferences into the common window after the prompts, the user does not need to do anything else. Eventually, the two figures will show up on the screen and the user will be able to see what the original coastline of the chosen city is compared to the new coastline of the same city based on the year they chose and the scenario chosen.

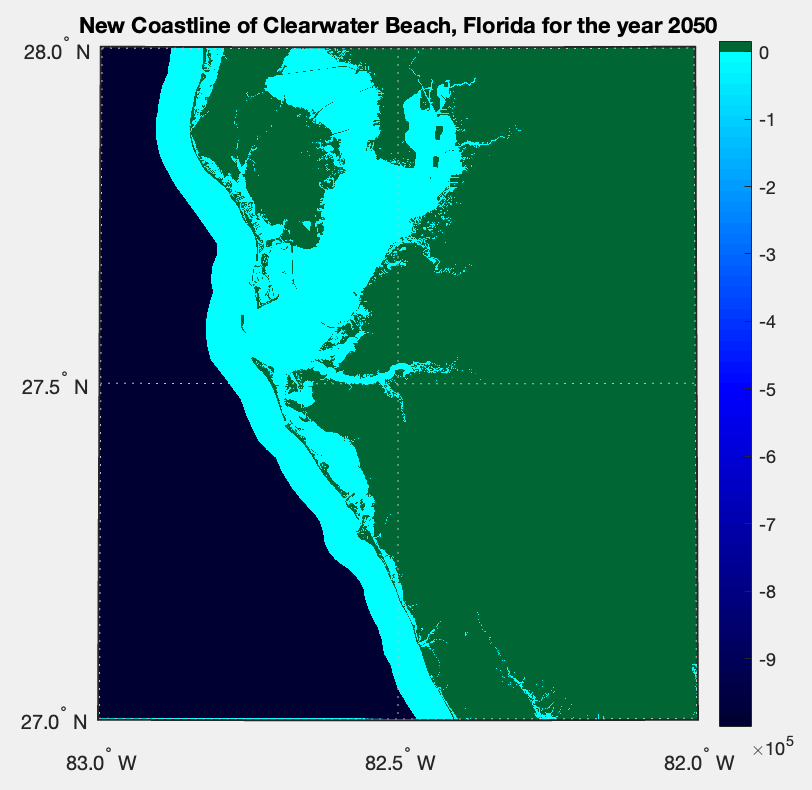
Figures 2.4 and 2.5 are examples of what the output would be if the user chose Clearwater Beach, Florida as the city, the year 2050 as the predicted year, and ‘3.0 - High’ as the scenario for the level of carbon emissions based on three meters of sea level rise by the year 2100. The light blue indicates shallow levels of water and the dark blue indicates deeper levels of water as shown by the colorbar on the figures. The green indicates land.



*Figure 2.3. User prompts and inputs in the command window of MatLab.*

**

*Figure 2.4. User’s view of the output of present elevation data.*

**

*Figure 2.5. User’s view of output from the user preferences.*

## Installation, setup, and support

To install the Coastline Predictor, start your device and open MatLab. Open the correct ‘finalfunction.m.’ Go to the link in the shared google drive and navigate to the coastline prediction data folder that will have the .csv file. Download the .csv file to your device. Drag and drop the .csv file into the current folder of MatLab, and run the script for the final function.

Once the script runs, the user will have to answer three different prompts by typing in the command window. The user will be guided on how to type correctly to each response. If the user does not type in a response correctly, the script will need to be rerun in order to work properly and have the correct output map of the new coastline.

If the user runs the Coastline Predictor too many times in a row, for multiple city locations, scenarios, or years, then the device is susceptible to crashing if the memory drive is too small. Devices with at least 16 GB can stand to run the script a few times before crashing.

# Operation of the Project (Stacia Kolodziejski)

## Operating Mode 1: Normal Operation

To operate the function, the user will need to interact with MatLab as follows:

1. The user will need to open Matlab along with the script for the final function that is provided for them. It is called ‘finalfunction.m.’
2. The users will run the function and wait for the prompts to appear in the command window.
3. The prompts that require an input from the users are: ‘Choose which city you would like to view the new coastline for:’, ‘Which level of CO2 emissions would you like to see and the level:’, and ‘What year would you like to view the new coastline for:’.

The only way for the user to interact with this function is to respond to the prompts as listed above.

If the user types in a response that does not correspond to the options, misspells a city name, does not choose a year in 10-year intervals from 2010 to 2100, or does not choose the correct level options that will be listed, the function will return an error and will not run.

## Operating Mode 2: Abnormal Operations

Abnormal operations include the collection of elevation data that is located too close to military bases. In this case, the user will be presented with a list of cities that are unable to be used with the function, so the user cannot choose these cities. If the user does choose one of these cities, the function will not operate normally and will not output the correct coastline map as it makes the original elevation data blurred and inaccurate.

To exit this abnormal state, the user can rerun the final function script and choose a new city that does work, meaning it is not listed on the abnormal cities list presented to the user.

Because the Coastline Predictor uses such large sets of data and has to run many times in order to view various different cities’ predicted coastlines, the user’s device has to have significant memory on the harddrive that can handle processing large amounts of data in MatLab. This is another abnormal state, and the way the user can exit this state is to reopen MatLab and run the final function script again.

# Technical Background (Wenyu (Jessica) Hu)

The goal of this product, Coastline Predictor, is to visualize what the coastline will look like up to 100 years into the future. The product consists of three main components: current elevation, sea level rise prediction, and analysis of the two to generate a map of future coastlines. The predictions of sea level rise are not made along with this product, instead it comes from credible sources. The National Oceanic and Atmospheric Administration (NOAA) is the source of data for predicted sea level rise. The current terrain elevation data comes from the United States Geological Survey (USGS).

Each of the components mentioned above is a MatLab function. The elevation data function will display a map of the current coastline of the desired region, as well as output a matrix containing the elevation values (in meters) of that region. The prediction data function will output a table of predicted sea level rise values (in centimeters) of multiple different regions under different scenarios every ten years up to the year 2150. The analysis function, named the final function, will prompt the user for the region, year, and scenario they wish to see and output a display of the current map and a display of the predicted coastline.

## Elevation Data

The code for the elevation part of this product references the open source MatLab code “terrain-elevation” posted by “spfrommer” on *GitHub*3. The outcome of spfrommer’s program is to display a three dimensional map of a designated area of the United States based on geographical coordinates. To do this, the program retrieves and downloads elevation data files from an Amazon Web Services (AWS) cloud file storage. Each file covers an area of 1x1 geographic degree (metric size varies depending on longitude) in 1 arc-second (around 30 meters) resolution. These data files are in *.tif* format and are sourced from the USGS. [[1]](#footnote-0)

The program would retrieve all the data files necessary to fill in the area requested by longitude and latitude limits, then combine those files and crop them to the requested limits. For example, if an area of [42.5 to 44] North and [71.5 to 70] West is requested, the program would retrieve elevation data files: [N42 to N43 x W72 to W71]; [N42 to N43 x W71 to W70]; [N43 to N44 x W72 to W71]; [N43 to N44 x W71 to W70]. Each of these data files can be processed into a matrix of the elevation values in meters. Then they are combined into one big matrix of [N42 to N44 x W72 to W70]. This big matrix is then cropped on its top and left side to [N42.5 to N44 x W71.5 to W70]. Once the elevation matrix is set, a three dimensional terrain map of the area will be displayed.

This product, Coastline Predictor, needs to visualize where the coastline is, thus a two dimensional map display is preferable. Once the elevation matrix of the desired region is created using the program by “spfrommer”, the function *geoshow()* is used to display a map of what that region currently looks like.

## Prediction Data

The prediction data of sea level rise is a *.csv* file sourced from NOAA2. The data provides predictions of sea level rise for every ten years, from the year 2020 to the year 2150. All predicted sea level rise values are in centimeters and are based on the sea level of the year 2005.

The .csv file contains not only numbers, but also headers for what the numbers in each row and column represent. There are also a few lines of notes explaining what the headers mean. For example, the headers “longitude” and “latitude” may seem obvious enough, however when examined closely, it appears that the longitude and latitude values listed for one region are the same throughout. As explained in the notes, this is because the longitude and latitude values listed are the center of a 1x1 geographic degree area.

Another important explanation of the header is the “scenarios”. The scenarios in the data file are listed as 0.3 m, 0.5 m, 1.0 m, 1.5 m, 2.0 m, and 2.5 m, which is the global mean sea level (GMSL) rise due t[[2]](#footnote-1)o low, intermediate-low, intermediate, intermediate- high, high, and extreme levels of ice sheets melt, respectively.

These notes are surely useful to understanding the data file, but they do get messed up when the *.csv* file is imported into MatLab for analysis. Therefore, these lines of notes must be erased before the prediction data can be processed in MatLab. To achieve this, MatLab was coded to simply skip the first few lines and start to import the data cells starting at the row where the headers start. Due to the fact that there are headers and other cells containing non-numeric values, the prediction data file would be imported into matlab as a *table*, instead of a *matrix* like the elevation data.

## Final Function - Analysis

The final function serves as the main function. This is where the elevation and prediction data functions will be called and analyzed based on the user’s choice of region, year, scenario (Ex: Boston, 2080, 1.0 m-intermediate ice sheet melt, respectively).

First, the prediction data function will be called to import the table of prediction data (Ex. called tableA). The program will then run through the “NOAA Name” column of tableA and extract into a new table (tableB) all the rows that are the name of the chosen region (Ex: Boston). The program will then run through the “Scenario” column in tableB and extract the row that matches the chosen scenario (Ex: 1.0). The program will then look for the header with the chosen year (Ex: 2080) and extract that value of predicted sea level rise. Since the elevation data being used is that of the current year, instead of the year 2005 referenced in the prediction data file, the predicted sea level rise of 2020 (closest to current year) must be subtracted from that of the chosen year. In the end, it will output the data points: latitude (Ex: 42.3539), longitude (Ex: -71.0503), predicted sea level rise value (Ex: 91 – 14 = 77).

Next, the elevation data function will be called. Since the longitude and latitude values extracted from the prediction data function are the centroid of a 1x1 geographical degree, the longitude and latitude limits will be the [longitude +/– 0.5] and [latitude +/– 0.5]. The elevation data function will operate as explained in *4.1* and output the elevation matrix of that region.

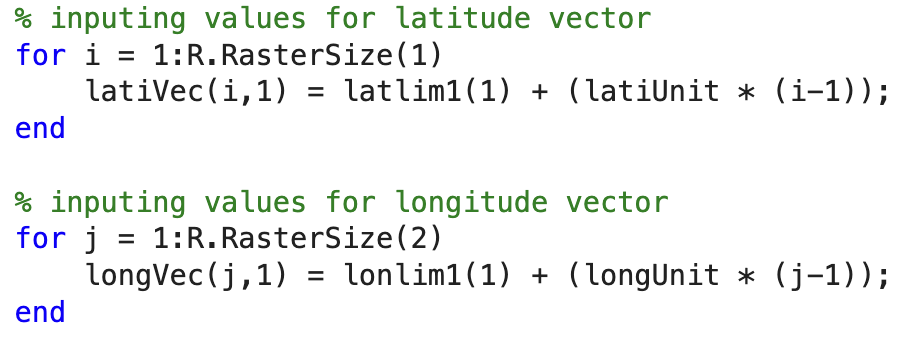
Finally, each cell of the elevation matrix will be subtracted by the predicted sea level rise value, which will result in a new matrix representing the predicted coastline of the chosen region in the chosen year under the chosen scenario.

The logic of the subtraction is that sea level rise can be considered as land level decrease. For example, when the sea level increases by 77cm, it means that 77cm in height of land will be covered by water. Thus, when measuring terrain (usually based on sea level), the land would have decreased by 77cm.

# Relevant Engineering Standards (Stacia Kolodziejski)

An engineering standard that is relevant to the Coastline Predictor is the Latitude/Longitude data standard as the function handles latitude and longitude entirely for numerous cities throughout the United States. The elevation data that is used for the function requires data for the latitude and longitude. The prediction data as well uses latitude and longitude to identify which region each set of data is for. Both of these datasets are required to contain latitude and longitude because they both are completely dependent on where in the United States the data is from and what it is used for as well.[[3]](#footnote-2)

The Coastline Predictor is also a function that is fully software and consists of coding throughout for each internal function and for the final function. There are specific engineering coding standards that are met in the Coastline Predictor as well. Our code is as simple as we could make it, there is little to no hard coding, the code is commented throughout to make it easy to detect errors within, and there are naming conventions that can be understood by an outsider. These are all advantages of implementing coding standards. The other coding standards that are used in the Coastline Predictor code are that we used internal functions for each task we needed to complete. Each function does a single task and the final function is what brings them all together to make the final output. There are indentations and paragraphs for each section of code to make it readable by the viewer.[[4]](#footnote-3) Figure 5.1 is an example of code demonstrating a few coding standards as discussed in the paragraph above.



*Figure 5.1. Paragraph of code from the elevation function to demonstrate coding standards.*

# Cost Breakdown (Stacia Kolodziejski)

| Project Costs for Production of Beta Version (Next Unit after Prototype) | | | | |
| --- | --- | --- | --- | --- |
| Item | Quantity | Description | Unit Cost | Extended Cost |
| 1 | 1 | MatLab\_R2022b | $0 | $0 |
| 2 | 1 | Elevation Data | $0 | $0 |
| 3 | 1 | Prediction Data | $0 | $0 |
| Beta Version-Total Cost | | | | $0 |

Budget Narrative

The Coastline Predictor is free for all users. All of the data used is open-sourced and does not require a subscription or any other form of payment to view, download, or use. The version of MatLab that is required to run the function is also free to any student or person associated with an institution. If a user does not have a user license through an institution, then they will have to pay for the MatLab software to use this function. The cost of MatLab without a license through an institution is $50., and the Mapping Toolbox is $30.

# Appendices (Stacia Kolodziejski)

## Appendix A - Specifications

| **Requirement** | **Value, Units** |
| --- | --- |
| Resolution of Maps | 10 meters |
| Sea Level Rise Prediction Units | Depth in meters |

## Appendix B – Team Information

Stacia Kolodziejski - B.S. in Electrical Engineering, 617-318-8158, [skolodz@bu.edu](mailto:skolodz@bu.edu)

Wenyu (Jessica) Hu - B.S. in Electrical Engineering, 617-858-0866, [wjhu@bu.edu](mailto:wjhu@bu.edu)

Saif Alblooshi - B.S. in Electrical Engineering, 617-309-8849, [sjalbloo@bu.edu](mailto:sjalbloo@bu.edu)

## Appendix C – Technical References

1. terrain-elevation, spfrommer, GitHub, Jan. 19, 2022, <https://github.com/spfrommer/terrain-elevation>
2. 2022 Sea Level Rise Technical Report, NOAA, Accessed March, 2023, <https://oceanservice.noaa.gov/hazards/sealevelrise/sealevelrise-data.html>
3. Latitude/Longitude Data Standard, U.S. EPA, Jan. 6, 2006. <https://www.epa.gov/sites/default/files/2015-06/documents/latlongstandard-v2a_10022014.pdf>
4. Coding Standards and Best Practices to Follow, Shreya Bose, Browser Stack, Feb. 2, 2021, <https://www.browserstack.com/guide/coding-standards-best-practices#:~:text=Practices%20To%20Follow-,What%20are%20Coding%20Standards%3F,sophisticated%20and%20highly%20functional%20code>.

1. terrain-elevation, spfrommer, GitHub, Jan. 19, 2022, <https://github.com/spfrommer/terrain-elevation> [↑](#footnote-ref-0)
2. 2022 Sea Level Rise Technical Report, NOAA, Accessed March, 2023, <https://oceanservice.noaa.gov/hazards/sealevelrise/sealevelrise-data.html> [↑](#footnote-ref-1)
3. Latitude/Longitude Data Standard, U.S. EPA, Jan. 6, 2006. <https://www.epa.gov/sites/default/files/2015-06/documents/latlongstandard-v2a_10022014.pdf> [↑](#footnote-ref-2)
4. Coding Standards and Best Practices to Follow, Shreya Bose, Browser Stack, Feb. 2, 2021, <https://www.browserstack.com/guide/coding-standards-best-practices#:~:text=Practices%20To%20Follow-,What%20are%20Coding%20Standards%3F,sophisticated%20and%20highly%20functional%20code>. [↑](#footnote-ref-3)