Web Based API Developer Portal/Simulator

By: Andrew Stanford, Jake Chappell, Austin Belt, and Jaren Provost

Business Requirements:

- **BR1: Marketing the Cheetah API**
  - The Cheetah API development portal will create more visibility for the API as well as clearly separate the API from the general Cheetah product for advertising and marketing usage of the API to clients.
- **BR2: Providing documentation for developers**
  - The Cheetah API development portal will allow developers to work with the API and understand it on their own; it will also save time that has previously been needed to explain the API and its capabilities/usages to developers or clients.

Actors:

- Cheetah API -> external system
- Developer -> system user
- External client -> system user
- Higher management -> stakeholder
- Maintaining developer -> admin

Use Cases 1:

- **UC1: Developer writing software**

  This would be a use case where a developer, data analytic, etc. is writing software that uses the Cheetah API's data and documentation or simply accessing data from the Cheetah API.

  Actors: Developer; Cheetah API

  Flow: Developer opens dev portal; developer logs in to private side of portal; developer uses the code workflows page and/or console/sandbox page to work with API data

  Business Requirement: BR(2)

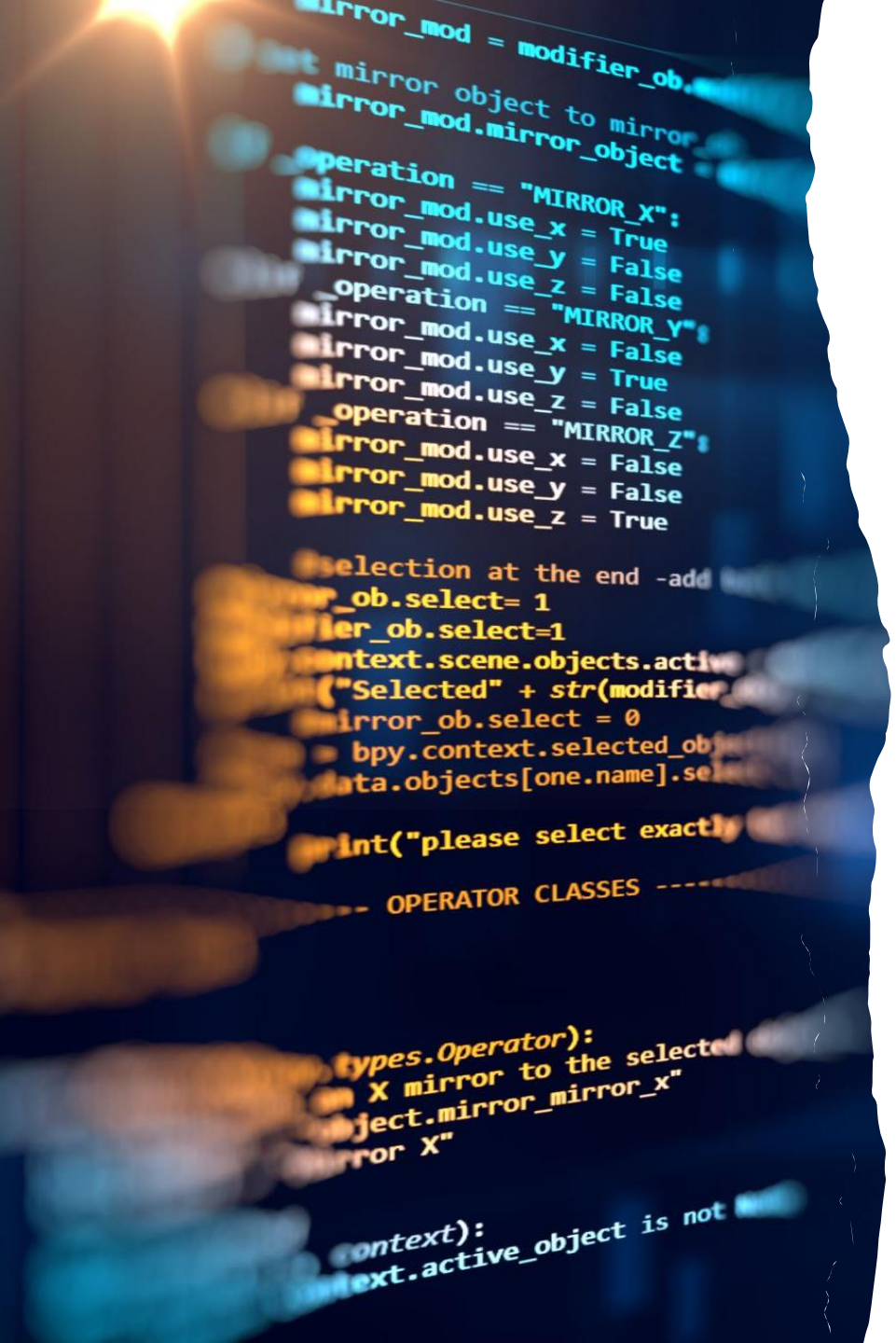
- **UC2: Higher management selling point**

  This would be a use case a where a president, stakeholder, etc. may be using the dev portal to see what the Cheetah API can do for them if they decided to buy access to the API.

  Actors: Higher management; Cheetah API

  Flow: Stakeholder opens dev portal; stakeholder may log in to private side if available to them; stakeholder browses main page/documentation page to see what capabilities the API has

  Business Requirement: BR(1)



Use Cases 2:

- UC3: External client using API

This would be a use case a where an external client using the Cheetah API may need some explanation on a workflow or API call they are trying to use.

Actors: External client; Cheetah API

Flow: Client opens dev portal; client may log in to private side if available to them; client browses main page/code workflow page to find explanations on the API calls and useful workflows they may be looking for

Business Requirement: BR(2)

- UC4: Admin managing dev portal

This would be a use case where an admin would log in to the dev portal to manage user access, documentation endpoints, general settings, etc.

Actors: Maintaining developer

Flow: Admin opens dev portal; admin logs in to private side of portal; admin opens a settings page to manage any available settings

Business Requirement: BR(2)

Requirements:

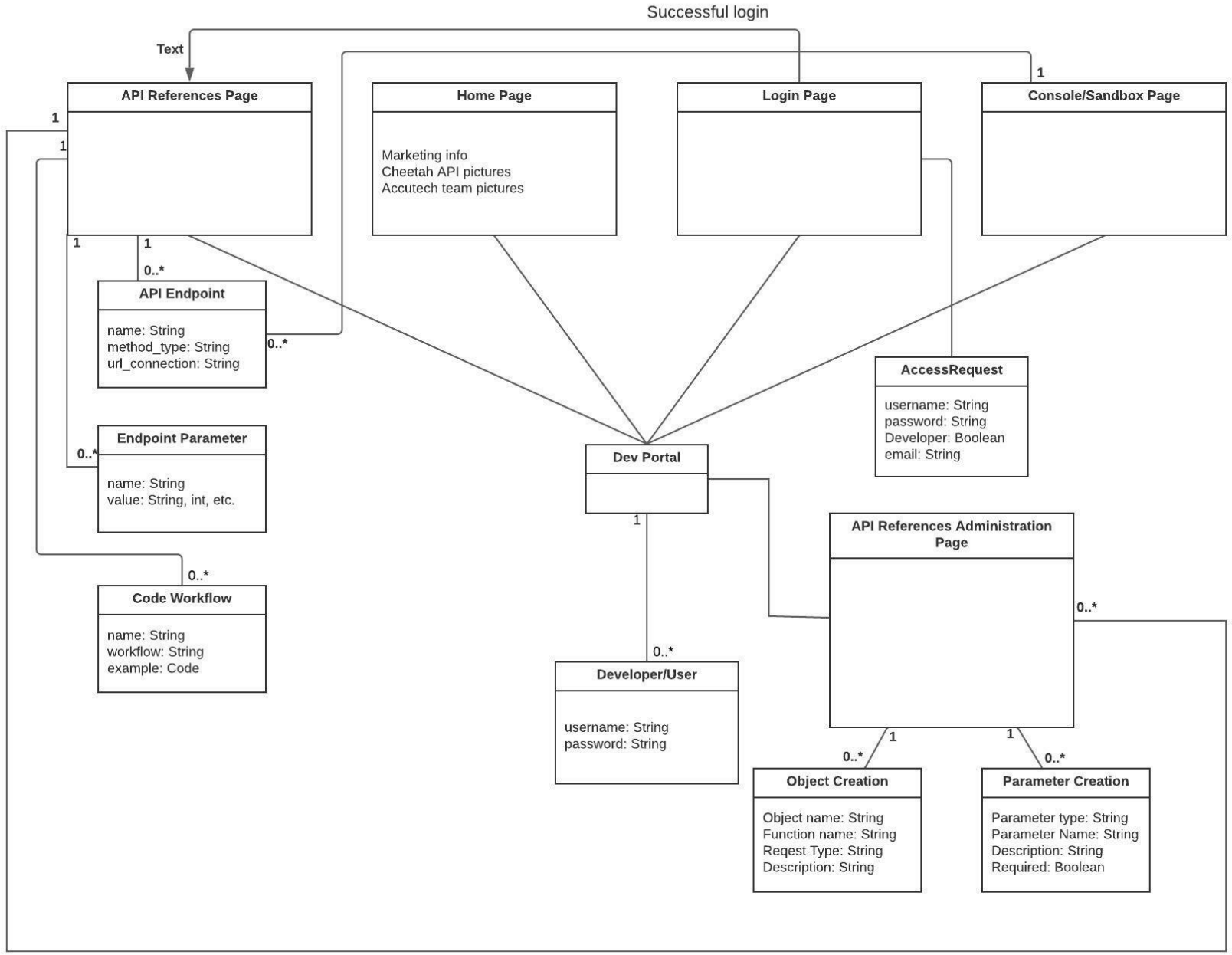
FUNCTIONAL

- **FR1:** The application shall have a home page with general information. [*Medium Priority*] (**BR1**)
- **FR2:** The application shall have a sign-in page to login as a specified user. [*Low Priority*] (**BR2**)
- **FR3:** The sign-in page shall allow users to request access to the documentation if they have not already been given access [*Medium Priority*] (**BR2**)
- **FR4:** The application shall have a page thanking the user for requesting access and giving them the option to return to the homepage until they gain access [*Medium Priority*] (**BR2**)
- **FR5:** The application shall have an API references page that documents each endpoint of the API along with the object that the endpoint works with [*High Priority*] (**BR2**)
- **FR6:** The API references page shall have a list of links to objects and endpoints included in the documentation that allows the user to navigate to where that object or endpoint is listed in the documentation [*Medium Priority*] (**BR2**)
- **FR7:** The application shall have a code workflow page to show common workflows for developers. [*High Priority*] (**BR2**)
- **FR8:** The application shall have an interactive console/sandbox page to test API calls with live data. [*High Priority*] (**BR2**)
- **FR9:** The console/sandbox page shall have capability to test API calls with live data and responses. [*High Priority*] (**BR2**)
- **FR10:** The application shall have an administration page that only those with administrators can visit. [*Medium Priority*] (**BR2**)
- **FR11:** The administration page shall let administrators create the form for each endpoint that will then be seen in the documentation. [*High Priority*] (**BR2**)
- **FR12:** The application will feature a navigation bar that allows the user to move to and from pages on the web portal. [*Medium Priority*] (**BR2**)

NON-FUNCTIONAL

- **NR1:** The application shall restrict access to private pages based on login status [*Low Priority*] (**BR2**)
- **NR2:** The application shall use electronic signatures to authorize user credentials for application login [*Low Priority*] (**BR2**)
- **NR3:** The console/sandbox page shall organize the API calls by object the call works with [*Low Priority*] (**BR2**)
- **NR4:** The application shall be designed the Accutech/Cheetah colors and logo [*Low Priority*] (**BR1**)
- **NR5:** Private access applications will be sent to an internal location for review before acceptance [*Medium Priority*] (**BR2**)

Domain Model:



Tech Stack:

Front End:

- Vue.js - Accutech currently uses Vue.js for their front-end web app functionality, and so we chose to keep consistent with the company here
- .NET Razor Pages - Accutech currently uses .NET Razor Pages for some functionality, and so we chose this to keep consistent with the company

Back End:

- .NET 5 - Accutech currently uses .NET for all back-end functionality, and so we chose this to keep consistent with the company
- ASP.NET MVC API - Accutech currently uses the MVC API for front to back end connections, and so we chose this to keep consistent
- C# open-source language - Accutech uses C# for their .NET programming, and so we chose to use it to keep consistent

Prototype

- Cheetah API Prototype Demonstration – YouTube
- Cheetah API Prototype – Figma

FIRST ITERATION FEATURES

- The first iteration will mainly focus on implementing the basis of the web pages and web page navigation.

- **FR1 –** The application shall have a home page with general information.
- **FR2 –** The application shall have a sign-in page to login as a specified user.
- **FR7 –** The application shall have a code workflow page to show common workflows for developers.
- **FR8 –** The application shall have an interactive console/sandbox page to test API calls with live data.
- **FR12 –** The application will feature a navigation bar that allows the user to move to and from pages on the web portal.

Mentor feedback:

- Suggested having a card view in the code samples/workflows page in order to keep things organized and readable – we adjusted our requirements accordingly

Client feedback:

- Client mentioned we may have too many requirements listed as high priority – suggested that we look over those priorities and set some as lower priorities, this was done
- Suggested we only look into very basic authentication and allowing a user to receive a login token – we adjusted our requirements accordingly

Jaren's 3-Legged German Sheppard

- Name: Sweetheart
- About:
  - Barks at almost everything
  - Acts Mean (It is for show)
  - Desires constant love and attention
  - Loves puddles / hates rain
    - Collapse in puddles upon impact
  - Favorite color is Red