Technology Essentials for Non-Tech Product Managers

API Cheat sheet

ProductHow.com

Introduction

What are APIs?	Are APIs and Web Services the same thing?
API (Application Programming Interface) is an intermediary that allows two applications to communicate with each other. This intermediary uses a combination of rules, protocols and tools to achieve this communication.	Well, there is a slight difference. Broadly speaking, web services are APIs that use a "network" to communicate (e.g. internet). There can be APIs that use other means of communication like DLL files, JAR files, etc. • Web Service is an API wrapped in HTTP • All web services are APIs but all APIs are not web services • Web service might not perform all the operations that an API would perform • A web service needs a network while an API doesn't need a network for its operation

Why every product manager should know the basics of APIs?

One of the key success factors for product managers is the ability to seamlessly translate engineer-speak to business-speak and vice versa. And APIs are something that both execs and engineers are going to talk about because of the business opportunities as well as the cool technology capabilities they offer. So it makes sense to understand the business implications as well as the essential terms that you are going to hear about on a frequent basis.

Business Scenarios concerning APIs

Scenario	Description	Benefits	Potential Pitfalls
Creating APIs for internal use	Develop the business functionalities in the form of re-usable APIs, available only to internal developers	Reduce development time and efforts as functionality once built, need not be built from scratch within every module	• Reusing a prebuilt API might not be that efficient as it would have been designed for a broader purpose • It takes conscious effort and push to write functionalities and then generalize them in the form of APIs. It needs to be ensured that the right slice of functionality is offered in the form of an API
Integrating with 3rd party APIs	Add functionality in the product by using services offered by 3rd parties. e.g. using Google Maps API to offer real time pizza tracking to your customers	• Quickly add complex functionalities to the product • High ROI as development efforts are relatively low	• Business risk as you are relying on an external party which might retire the API or in some cases might just shut shop • Cost implications might become substantial as the user base grow
Building & exposing APIs to customers	Build a functionality that you believe would be useful to other developers & businesses and expose them. For some companies, APIs are the main business while for some it can be an addition to their core offering	An excellent source of recurring revenue which grows as your customers grow	If APIs are not your core offering, due care needs to taken that you are not spending considerable time and efforts in developing APIs that might not actually find many takers in the market. It might be the case that your product is doing well, but the related APIs don't find many takers

Technology Concepts

An analogy to remember: **"Think of a restaurant. The menu is the API, your order is the request, the food from the kitchen is your response"**

10 Essential Technical Terms to Know

Term	Description	The restaurant analogy
Requests & Responses	You make a request in a pre-defined format and the API comes back with a response, again in a pre-defined format	Request is the order placed by you based on the menu and the response is the food received by you as described in the menu
REST	Short for Representational state transfer. It describes a set of standard, universally accepted architectural principles and characteristics	Imagine an restaurant operations expert standardizing how the menu would look like, how to ask for waiter's attention, how the waiter would respond, etc
HTTP Methods	They define what type of tasks you can achieve through an API. Most commonly used HTTP methods include POST, GET, PUT, PATCH, DELETE	Imagine the standards put in place by the restaurant operations expert standardize what are the expected requests you can make to the waiter – take food order, check status of order, cancel an ordered dish, etc. Really helpful to avoid absurd requests like asking the waiter to build a house for you while you wait for your order!
Endpoints	Connection point which accepts requests to access resources on an API. An endpoint needs to be 'exposed' so that it can be called.	Think of it as the restaurant's address and its door location. You need to know this information before you can enter and place an order
API documentation	It lays down the rules of the 'contract' and helps the engineering team understand what functionality is available by using different API calls.	Think of it as the restaurant policy usually printed and pinned on the restaurant wall/ website
API Calls	We refer to the requests to APIs as API calls. API calls involve hitting an endpoint with the expectation that the API will respond with the information you need.	The entire process of you stepping into the restaurant, sitting down at a table and placing your order with the waiter.
Payloads	It refers specifically to the meaningful information in a given set of data. Not everything in an APIs response might be useful, so the term payload is used to refer the useful stuff among the entire response.	Think of it as the meat on the plate. While garnishing and other stuff on the plate might be appreciated, you might not care about them much when you are hungry.
Response Codes	Every time you get a response from an API, it comes with its own response code, which is simply a number with a meaning attached to it.	Think of it is as a note from the kitchen that comes along with your order. It might say "200 - All well, enjoy your food" or "500 - the stove is on fire, please expect a delay in your order" or "404 - you ordered something that is not available"
Headers	Headers are additional useful bits of information which are sent and received along with your HTTP methods. Some of the common info includes – Authorization Credentials, Content Type, Date, etc	Request Header: The waiter notes the table number on top of your order slip Response Header: Chef mentions that the lid should not be removed by the waiter when placing the dish on the table
Authentication	In order to establish your connection with the API, you need to tell the API. that you're a genuine user who has the necessary privileges to access the API. In order to do this, you need to access APIs using an authentication token. Oauth is a popular authorization framework used for authentication of users.	Imagine the restaurant is open only for members. You need your membership cards and credentials to enter or receive service from the restaurant