w4School

All for The University of Sydney Students

WEBSITE DEVELOPMENT PROJECT

RE03_GROUP4

BINGYAN CHEN|HAOCHEN ZHANG|KAIJUN PENG|LIUYU|JACKSON WANG
480049942| 490169971| 490441372| 490095315| 480342832

Contents

Introduction

In this project, various techniques and tools are utilized to understand and develop an outstanding project. Below are the detailed descriptions for each tool.

Figma is a prototype design tool that runs on the cloud. In theory, as long as the internet is connected, it is available to anyone on any device. It has several advantages compared to other prototype design tools. Firstly, it is fast. The biggest pain for other tools is that they are easy to crash if the design files are slightly larger. However, our group has never encountered such a serious situation. Besides, it supports multiplayer online editing at the same time. This is almost the most important function of Figma. Since this is a group project, every team member needs to contribute to the design. It can subvert the original digital product design process and mode, which disclose the originally invisible design operations. It can greatly reduce the cost of communication within each team member and can quickly collect feedback. Lastly, Figma combines feathers from other tools. Figma is more than just a prototype design tool, its functions have almost covered all important steps in digital product design. For instance, our team can use Figma to complete the design, then make an interactive prototype and collect feedback through evaluations. These steps can be completed only by relying on Figma.

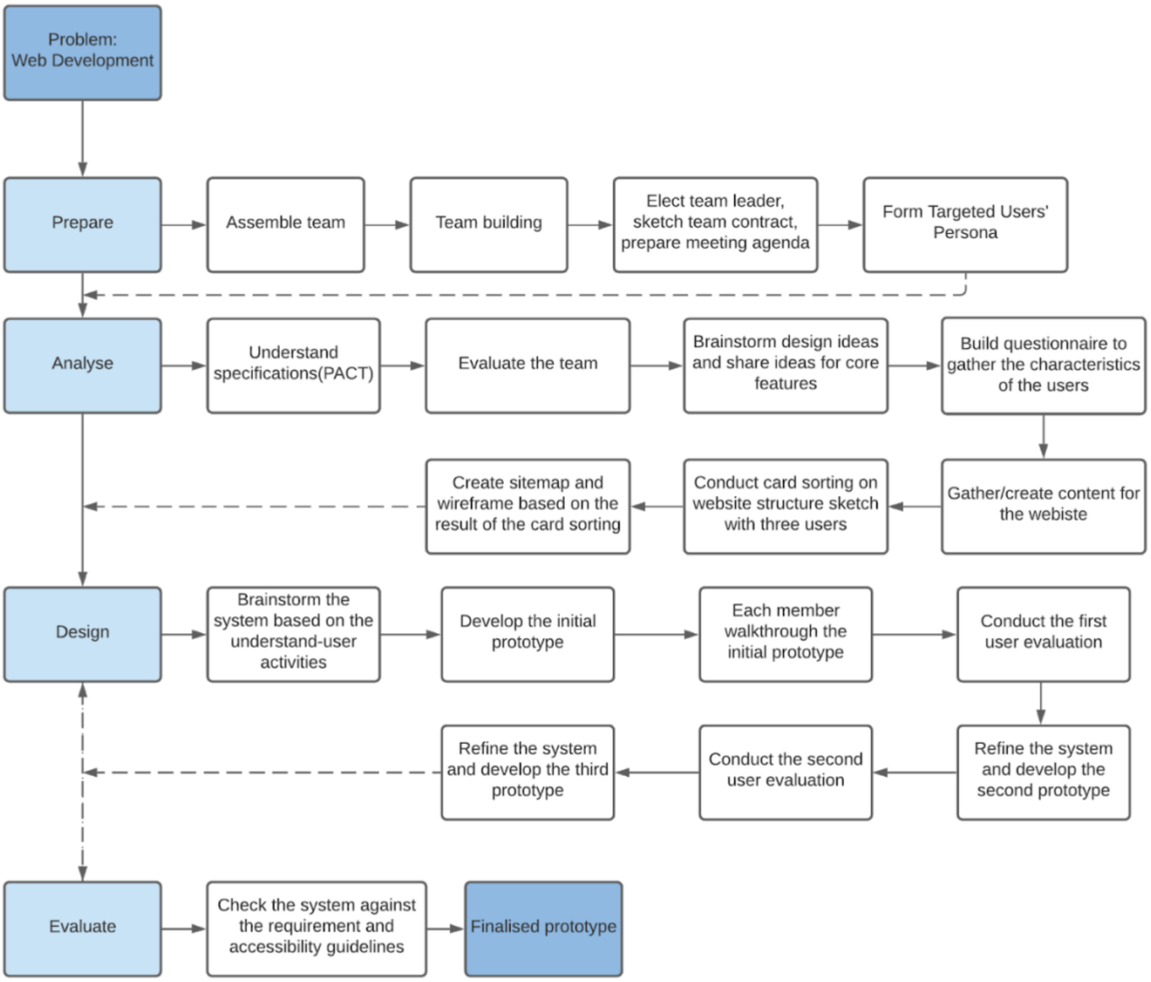
As for the communications between team members, our team relies on WeChat and Zoom. WeChat supports various platforms (e.g. IOS, Android. etc.) to quickly send text and voice messages through the network. Since all our team members are overseas now, WeChat is the best option for daily communication and getting quick responses. Our team communicates daily through WeChat and asks for feedback and answers if we are not sure about certain questions. Zoom supports free cloud video conferencing both on PC and mobile platforms among diverse people. We use Zoom for our weekly meetings to communicate about what progress to be made and any questions to be solved.

Google Docs is an online document that can be edited by multiple people at the same time, supporting online Word or PDF, which can be viewed and modified anytime, anywhere on the computer, iPad, and other types of devices. The document can be viewed and edited as long as the webpage is opened and saved in the cloud in real-time. Our team uses it to write documentation such as meeting minutes. Google Forms is a survey management application that lets users investigate and collect information from others. Besides, Google Forms also supports multi-user collaboration and sharing functions. In this project, before developing actual software, our team launches a survey to investigate who are the target users along with their characteristics for the software and what functions are useful to them. Google Chrome is a web browser which is used to search and gather relevant information from various sources.

Goodnotes and Notability are two software tools for taking notes on iPad. Since all team members have an iPad, we select this software to take some quick notes during tutorials and meetings. Besides, some initial design ideas are drawn using the software.

Overview

Below is the lifecycle diagram of our design process.


Problem: Web Development
Prepare
Assemble team
Team building
Elect team leader, sketch team contract, prepare meeting agenda
Form Targeted Users' Persona
Analyse
Understand specifications(PACT)
Evaluate the team
Brainstorm design ideas and share ideas for core features
Build questionnaire to gather the characteristics of the users
Gather/create content for the webiste
Conduct card sorting on website structure sketch with three users
Create sitemap and wireframe based on the result of the card sorting
Design
Brainstorm the system based on the understand-user activities
Develop the initial prototype
Each member walkthrough the initial prototype
Conduct the first user evaluation
Refine the system and develop the second prototype
Conduct the second user evaluation
Refine the system and develop the third prototype
Evaluate
Check the system against the requirement and accessibility guidelines
Finalised prototype

And the following is the task with its appendix reference.

Task	Appendix Reference
PACT analysis	-
Team persona	-
Investigation protocol	-
Implications of PACT analysis	-
Card sorting	D
Wireframe	E
First usability testing	F
Second usability testing	G
Graphical representation of group's journey	I

Our website design procedures are designing our persona first, then we make a survey. After gathering the data, we design the website wireframe and sitemap. Then we finish our prototype and refine the prototype iteratively according to the comments from user evaluation.

Next, we will introduce each process undertaken by our group in order.

Phase 1 User Investigation

To better under the users' need, the team apply the PACT analysis technique and it could be found in Appendix H for further reference.

PACT helps the team in the following phases in several ways. Since the design of UX is related to human characteristics in many ways. PACT analysis can help designers to get a full map of what factor of human will impact the interaction between human and computer(system).

People differ in terms of physical and psychological qualities, as well as in how they use systems. Activities differ in terms of timing, complexity, safety, and the type of content required. Physical, legal, and organizational characteristics of contexts differ. The input, output, communication, and content that technologies support is all different.

The established PACT result reminds the team of the true user needs in the broad sense. It ensures the team is on the right track when the team brings up in-depth designs or there is a dispute.

Team Evaluation

Team-based persona

Attribute	**Value**
Name	John Columbia
Age	22 (Second Year)
Gender	Male
Background	Grew up in ACT

Course (Degree)	Bachelor of Advanced Computing
Knowledge/Interests	Video games, Diving, racing car, Singing
Skills	R studio, Java, Python, SQL
Goals	HD mark Web design/development

User Evaluation

We use the google form to make our questionnaire. Then we distributed the questionnaire across our network to analyse and learn our target users' group. The questionnaire was launched on March 13th, and we were unable to complete it until March 18th due to continuous submissions. Additionally, the project's key criterion is to teach "every University of Sydney student," and data reliability has a good association with the size of the experimental community, so those two considerations are also taken into account. Further reference could be found in Appendix C.

The team developed and accepted the questionnaire. It is made up of demographic questions, such as: e.g. How old are you, Do you have IT background, and behaviour questions e.g., what are you expecting from the website? These behaviour questions would help the team to design to fit the user's needs.

The results show that 71.3 percent of our respondents have an IT history, while 28.3 percent do not, which is possibly biased because we only sent the questionnaire to friends and acquaintances. Some respondents shared curiosity in Python bottle, JavaScript, and HTML tutorials, whilst others lamented their lack of CSS and Framework awareness. In terms of content delivery, video plus text came out on top by a wide margin. Users come to our website for a variety of reasons, which reminds the team to diversify the content. Finally, 79.2 percent of respondents requested multilingual assistance.

The team realized questions may have left open-ended responses for respondents when reviewing, because the choices given could be limited, especially in the question, "What is your purpose for using our website?" There should be more demographic questions, not just technological, behaviour questions, to get a better picture of users' profiles.

The user persona below is generated based on the summary of the collected submissions.

Survey-based User persona

Attribute	**Value**

Name	Stella Lau (she/her)
Age	21 (Second Year)
Gender	Female
Background	Born and grew up in Sydney
Reason for choosing USYD	Like all the societies on campus
Likes about USYD	Water bubblers
Dislikes about USYD	People sleeping in the library
Program of study	Advanced Computing (Data Science, Computer Science)
Favourite unit of study	INFO1113
Personal challenges	Long commute
Career goal	Financially independent + self-employed
Plans after graduation	Full-time work
Goals	Prepare for a job, learn to develop websites and be proficient
IT Background	Yes
Expectation about the website	Python bottle
Preferred learning methods	Text and video
Special requirement	Multi-language support

Phase 2 Navigation design

Card sorting

We divided our sort into two parts, one of them is our website-function-oriented card sorting, another one is the card sorting about programming knowledge.

The programming knowledge part achieved 100% agreement about the card sorting. The specified knowledge about HTML, Python Bottle, CSS and JavaScript will be assigned to HTML module, Python Bottle module, CSS module and JavaScript module respectively.

Closed card sorting
Open card sorting

In order to keep consistency on the overall website, the team decided to have the same heading for our website.

We brainstormed a list of question/content that users may have whilst using the website. The questions are divided into two section, one open card sorting and one closed card sorting. The open card sorting is designed to observe that how users perceive the different course content e.g., python bottle, python syntax, JavaScript structure and JavaScript syntax. The closed card sorting is designed to categorize the website content and functionality e.g., FAQs, profile page, discussion channel, in the perspective of users, to construct the navigation system for the website.

Information architectures

After having the website navigation, the team started our design of wireframe and sitemap. We discussed the functions and pages we need in our web, and that consists of account page, course page, forum page, feeds page, message page, log in(up) and so on. Then everyone draws one- or two-pages' wireframe. In the next zoom meeting, everyone shows their wireframe, then combines the advantages of each wireframe and draws a final one together. Sitemap is based on the wireframe, each page is a block in the sitemap, and then discusses the connection of each block.

The sitemap and wireframe are the exact reflection of user needs. It is designed based on the result of user card sorting and brought a sketchy layout of the website. Besides, further iterative improvement and implementation is needed.

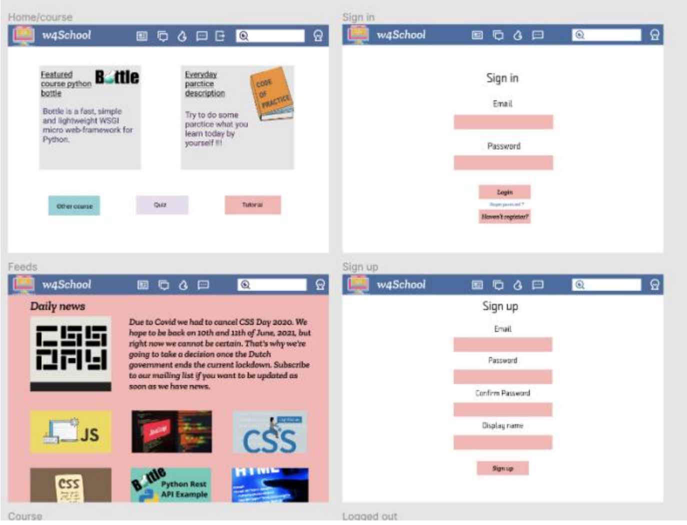
Phase 3 Iterative design

As the final output of the whole design process, the team decided to conduct iterative prototype design, which consists of three iterations of prototype and two usability testing in between. Before we design the initial prototype, each team member collects good design websites, and we discuss the style that benefits us. And style final decided is simplicity and the colour should be light, since the light colour and the contrast design could make the user feel comfortable. Then everyone draws the same pages as the wireframe. Course pages are the core pages. The resources we use in the website are picked from the internet, and further reference list could be found in Appendix A. The team constructed a protocol to ensure the quality and reliability, which could help users study easier.

Usability testing is a crucial part of the project, which could effectively evaluate the quality of the prototype. The team as designers, sometimes might have some bias to our own prototypes. We feel everything is perfect and can't find some defects. But when participants run our task, the problems can emerge immediately. Thus, we try to make the task complete which could include all the operations our prototype can. And after the reflection has been collected, the team will try to fix the complained problems and adopt the proper and reasonable suggestions into the next iteration of the prototype.

Prototype 1st iteration

We adopt the same style as Ed. If we want to prove user usability, Simplified operation is needed. So, we design a main bar, it likes navigation. Users can view any core pages by clicking the icon in the main bar. And we draw the main bar together and add some simple icons. To clean the ambiguity of the icon, we group up the words and icon, and the main bar includes account, course, forum, feeds, message, and our website symbol. At this stage, we focused on implementing all required student functions. And in any page, we only add the necessary image and words to reduce the time to find information.



Featured course python bottle

Bottle is a fast, simple and lightweight WSGI micro web-framework for Python.

Everyday parctice description

Try to do some parctice what you learn today by yourself !!!

Daily news

Due to Covid we had to cancel CSS Day 2020. We hope to be back on 10th and 11th of June, 2021, but right now we cannot be certain. That's why we're going to take a decision once the Dutch government ends the current lockdown. Subscribe to our mailing list if you want to be updated as soon as we have news.

Initial prototype user evaluation result

We as a team conducted cognitive walkthrough after designing. We cannot identify any issues since the whole system was all designed by ourselves, but it turned out as problematic design and logic in the usability testing.

First usability testing

The responses have been summarized in the table below.

	Pages	**User Response (Any issues?)**	**Additional Comments**
1.	Sign up	Better change words “email” to “username”	
2.	Log in	None	Email login might be too long, we could use unikey
3.	View Bottle course	None	The font too big, Bottle image could be smaller
4.	View JavaScript course	The size of the font in the menu is too small	· Image could be set in higher resolution · It will be better if the menu can be hided. · The button of the course like a course schedule, poor pick
5.	View HTML practice	None	There are only questions without any tutorials.
6.	View Python practice	The size of the font in the menu is too small	
7.	Message a user	Can’t use the message since I can’t type a word or DM with others and cannot make a group talk.	Should improve this function and make DM achievable
8.	View message history	Task set inappropriately since there is no message listed	

9.	View Forum	None	Filter is not clear
10.	Manage profile	I can’t manage my profile, but I can check my profile	• Font could be improved • Allow user to manage profile
11.	View Feeds	I can see the feeds, but I cannot click on them	• Allow users to check multiple feeds • The button of feeds is too ambiguous

Interview

General opinions

- Other course in the course page is confusing, there are four links to one page. They can be combined, which will save place for another potential functional button. Featured courses other courses could be put in one page with featured courses prioritized.
- Course Page needed be improved, cannot find what I need in time and effectively.
- Pretty good about log in and create account and also seeing course but the other functions are seemed not prepared enough.

Suggestions

- Subscribe button in Feeds.
- Better aesthetic design
- Back button

In the summary above, most of the participants addressed that the login, sign up page are well designed. The main issue is that the *other course* button in the course page, which was complained by the participants the most. Other issues include the unimplemented functionality in the message page, image resolution, font issues and other aesthetics issues.

Although this, the participants commented highly on the messaging image, audio chat and video chat functions, but that would be just some sweet only in prototype, whether implement was yet to decide.

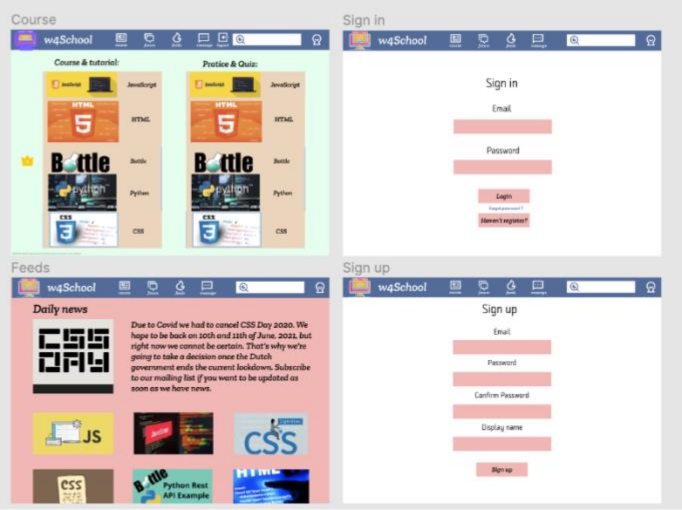
Whilst we gather the feedback of the first prototype, some of the participants advised that go-back button in course page, which will be considered in the actual implementation but not in the prototype iteration as its practicality. Another insightful suggestion is share function in feeds page. That will be considered in the implementation as the project prohibits the use of CDN and external resources.

Based on the feedback we got above, the following tweak will be applied in the next prototype:

- Redesigned website header with text descriptions.
- Redesigned course with clearer layout.
- More animation on messaging page.
- ‘Update password successfully’ pop-up window after update password.

Prototype 2nd iteration

After the first user evaluation, we discuss the comment of the participant, and adopt some good ideas. The obvious change is grouping up with words and icons. In participant comments, most of the participants confuse the icon. When they do the task, they can’t identify the forum and message. So, we add the words to help people view different pages easier. The first iterative course design is so weak, so we also upgrade the course pages. In the first iterative course, if we want to view a course, we have to click the button on the home page to course pages. But in this iteration, users don’t need this process anymore, it can save some time.


Due to Covid we had to cancel CSS Day 2020. We hope to be back on 10th and 11th of June, 2021, but right now we cannot be certain. That's why we're going to take a decision once the Dutch government ends the current lockdown. Subscribe to our mailing list if you want to be updated as soon as we have news.

Second usability testing

The second usability testing template inherited the test plan from the first usability testing. And it is summarized below.

	Pages	**User Response (Any issues?)**	**Additional Comments**
1.	Sign up	None	· It is good. But what does a resetting password need? Feel like only a security question is not enough…. · Font size seems too large
2.	Log in	None	
3.	View Bottle course	None	I have looked at all the courses and practices. It is well formatted.
4.	View JavaScript course	None	Navigation has the same background colour with content
5.	View HTML practice	None	'Practice & course' could be larger to draw more attention
6.	View Python practice	None	
7.	Message a user	Unable to type in anything yet	· Add friend button in channel page could be improved · Users' icon can be shown in the contact's list. · Page 7 is so big that it cannot display on my side, even when I tried to zoom out.

8.	View message history	None	
9.	View Forum	The layout is good, but not able to switch to other forum windows	The comment box is a bit small compare to the content part. Users may feel uncomfortable if they want to type in long contents.
10.	Manage profile	The 'upload pic' option is not available	· Has a little issue on locating · The 'Moments' section is unmeaningful, no explanation. · Safety should in the setting
11.	View Feeds	None	Need a record and load button to feel more natural

Interview

General opinions

- The font size of pages is not consistent, may cause users hard to read.
- The contents/icons of course & quiz home page seems a bit overlapping. Maybe try to use only one icon for each programming language and give choices for course and quiz under each language.
- The icons of user can be added to the contacts list of chatting window, so that users can distinguish people easier.
- Navigation has the same background colour with content some backgrounds colour should be changed, light colour could be better

Suggestions

- Share function with website screenshot downloaded or Website bookmark

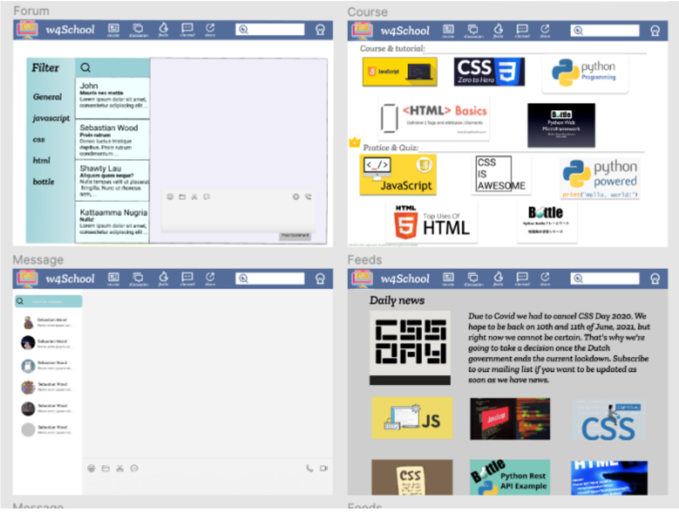
The second usability testing look in depth of the second prototype. The positive comment significantly increases compare to the first prototype. Overall, the participants are satisfied with the intuitive navigation bar, course content page and login page. Besides, the participants pointed the design flaws in helping text, buttons, account profile pictures, and consistence.

Based on the above usability testing, the team decided to have the following improvement in the next prototype:

- Improve interaction in the forum and messaging page
- Redesign course page
- Refine overall aesthetics setting includes colour theme and page layout
- Enrich interaction in admin page

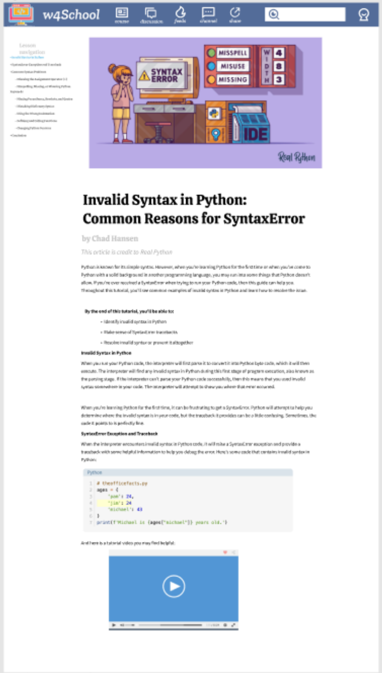
Prototype 3rd iteration

And this is our final version prototype, we upgrade many details to improve beauty. And add many pages which can show functions more clearly. This prototype also clears many mass parts.



Due to Covid we had to cancel CSS Day 2020. We hope to be back on 10th and 11th of June, 2021, but right now we cannot be certain. That's why we're going to take a decision once the Dutch government ends the current lockdown. Subscribe to our mailing list if you want to be updated as soon as we have news.

Python Course



w4School course discussion feeds channel share

Lesson navigation

Invalid Syntax in Python: Common Reasons for SyntaxError

by Chad Hansen

This article is credit to Real Python

Python is known for its simple syntax. However, when you're learning Python for the first time or when you've come to Python with a solid background in another programming language, you may run into some things that Python doesn't allow. If you've ever received a SyntaxError when trying to run your Python code, then this guide can help you. Throughout this tutorial, you'll see common examples of invalid syntax in Python and learn how to resolve the issue.

By the end of this tutorial, you'll be able to:

- Identify invalid syntax in Python
- Make sense of SyntaxError tracebacks
- Resolve invalid syntax or prevent it altogether

Invalid Syntax in Python

When you run your Python code, the interpreter will first parse it to convert it into Python byte code, which it will then execute. The interpreter will find any invalid syntax in Python during this first stage of program execution, also known as the parsing stage. If the interpreter can't parse your Python code successfully, then this means that you used invalid syntax somewhere in your code. The interpreter will attempt to show you where that error occurred.

When you're learning Python for the first time, it can be frustrating to get a SyntaxError. Python will attempt to help you determine where the invalid syntax is in your code, but the traceback it provides can be a little confusing. Sometimes, the code it points to is perfectly fine.

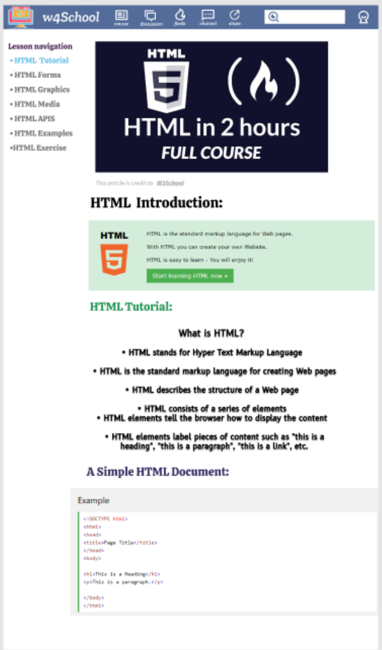
SyntaxError Exception and Traceback

When the interpreter encounters invalid syntax in Python code, it will raise a SyntaxError exception and provide a traceback with some helpful information to help you debug the error. Here's some code that contains invalid syntax in Python:

Python
```
# theofficefacts.py
ages = {
    'pam': 24,
    'jim': 24
    'michael': 43
}
print(f'Michael is {ages["michael"]} years old.')
```

And here is a tutorial video you may find helpful:

HTML Course



w4School course discussion feeds channel share

Lesson navigation

- HTML Tutorial
- HTML Forms
- HTML Graphics
- HTML Media
- HTML APIS
- HTML Examples
- HTML Exercise

HTML in 2 hours FULL COURSE

This article is credit to W3School

HTML Introduction:

HTML is the standard markup language for Web pages.

With HTML you can create your own Website.

HTML is easy to learn - You will enjoy it!

Start learning HTML now »

HTML Tutorial:

What is HTML?

- **HTML stands for Hyper Text Markup Language**
- **HTML is the standard markup language for creating Web pages**
- **HTML describes the structure of a Web page**
- **HTML consists of a series of elements**
- **HTML elements tell the browser how to display the content**
- **HTML elements label pieces of content such as "this is a heading", "this is a paragraph", "this is a link", etc.**

A Simple HTML Document:

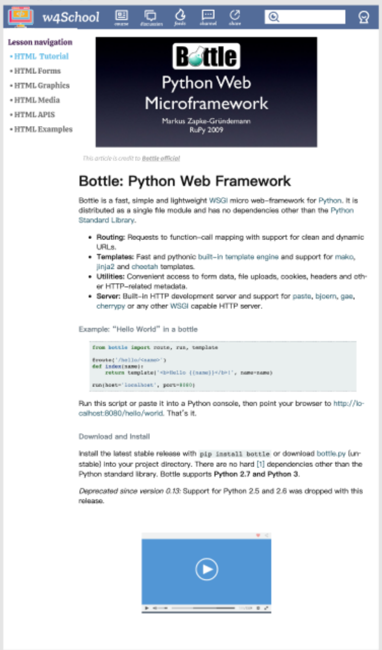
Example

```
<!DOCTYPE html>
<html>
<head>
<title>Page Title</title>
</head>
<body>

<h1>This is a Heading</h1>
<p>This is a paragraph.</p>

</body>
</html>
```

Bottle Course



w4School course discussion feeds channel share

Lesson navigation

- HTML Tutorial
- HTML Forms
- HTML Graphics
- HTML Media
- HTML APIS
- HTML Examples

Bottle Python Web Microframework
Markus Zapke-Gründemann
RuPy 2009

This article is credit to Bottle official

Bottle: Python Web Framework

Bottle is a fast, simple and lightweight WSGI micro web-framework for Python. It is distributed as a single file module and has no dependencies other than the Python Standard Library.

- **Routing:** Requests to function-call mapping with support for clean and dynamic URLs.
- **Templates:** Fast and pythonic built-in template engine and support for mako, jinja2 and cheetah templates.
- **Utilities:** Convenient access to form data, file uploads, cookies, headers and other HTTP-related metadata.
- **Server:** Built-in HTTP development server and support for paste, bjoern, gae, cherrypy or any other WSGI capable HTTP server.

Example: "Hello World" in a bottle

```
from bottle import route, run, template

@route('/hello/<name>')
def index(name):
    return template('<b>Hello {{name}}</b>!', name=name)

run(host='localhost', port=8080)
```

Run this script or paste it into a Python console, then point your browser to http://localhost:8080/hello/world. That's it.

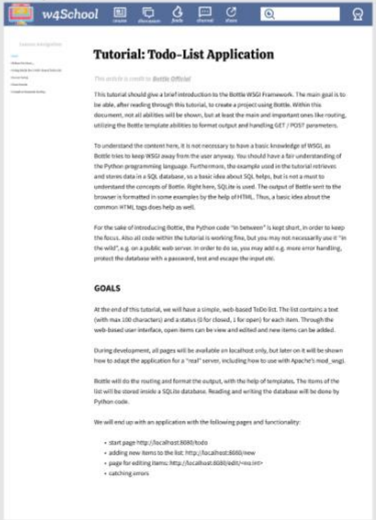
Download and Install

Install the latest stable release with `pip install bottle` or download bottle.py (unstable) into your project directory. There are no hard [1] dependencies other than the Python standard library. Bottle supports **Python 2.7 and Python 3**.

Deprecated since version 0.13: Support for Python 2.5 and 2.6 was dropped with this release.

Python practice

Bottle practice



w4School course discussion feeds channel share

Lesson navigation

Tutorial: Todo-List Application

This article is credit to Bottle Official

This tutorial should give a brief introduction to the Bottle WSGI Framework. The main goal is to be able, after reading through this tutorial, to create a project using Bottle. Within this document, not all abilities will be shown, but at least the main and important ones like routing, utilizing the Bottle template abilities to format output and handling GET / POST parameters.

To understand the content here, it is not necessary to have a basic knowledge of WSGI, as Bottle tries to keep WSGI away from the user anyway. You should have a fair understanding of the Python programming language. Furthermore, the example used in the tutorial retrieves and stores data in a SQL database, so a basic idea about SQL helps, but is not a must to understand the concepts of Bottle. Right here, SQLite is used. The output of Bottle sent to the browser is formatted in some examples by the help of HTML. Thus, a basic idea about the common HTML tags does help as well.

For the sake of introducing Bottle, the Python code "in between" is kept short, in order to keep the focus. Also all code within the tutorial is working fine, but you may not necessarily use it "in the wild", e.g. on a public web server. In order to do so, you may add e.g. more error handling, protect the database with a password, test and escape the input etc.

GOALS

At the end of this tutorial, we will have a simple, web-based ToDo list. The list contains a text (with max 100 characters) and a status (0 for closed, 1 for open) for each item. Through the web-based user interface, open items can be view and edited and new items can be added.

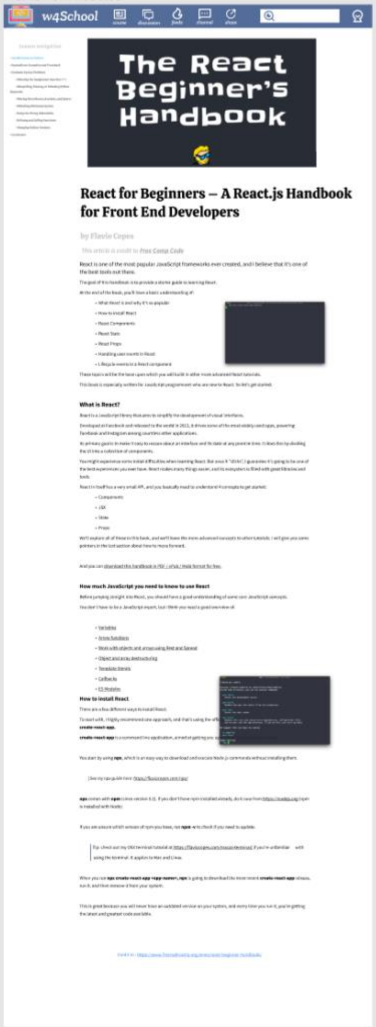
During development, all pages will be available on localhost only, but later on it will be shown how to adapt the application for a "real" server, including how to use with Apache's mod_wsgi.

Bottle will do the routing and format the output, with the help of templates. The items of the list will be stored inside a SQLite database. Reading and writing the database will be done by Python code.

We will end up with an application with the following pages and functionality:

- start page http://localhost:8080/todo
- adding new items to the list: http://localhost:8080/new
- page for editing items: http://localhost:8080/edit/<no:int>
- catching errors

Framework ReactJs



w4School course discussion feeds channel share

Lesson navigation

The React Beginner's Handbook

React for Beginners – A React.js Handbook for Front End Developers

by Flavio Copes

This article is credit to Free Camp Code

React is one of the most popular JavaScript frameworks ever created, and I believe that it's one of the best tools out there.

The goal of this handbook is to provide a starter guide to learning React.

At the end of the book, you'll have a basic understanding of:

- What React is and why it's so popular
- How to install React
- React Components
- React State
- React Props
- Handling user events in React
- Lifecycle events in a React component

These topics will be the base upon which you will build in other more advanced React tutorials.

This book is especially written for JavaScript programmers who are new to React. So let's get started.

What is React?

React is a JavaScript library that aims to simplify the development of visual interfaces.

Developed at Facebook and released to the world in 2013, it drives some of the most widely used apps, powering Facebook and Instagram among countless other applications.

Its primary goal is to make it easy to reason about an interface and its state at any point in time. It does this by dividing the UI into a collection of components.

You might experience some initial difficulties when learning React. But once it "clicks", I guarantee it's going to be one of the best experiences you ever have. React makes many things easier, and its ecosystem is filled with great libraries and tools.

React in itself has a very small API, and you basically need to understand 4 concepts to get started:

- Components
- JSX
- State
- Props

We'll explore all of these in this book, and we'll leave the more advanced concepts to other tutorials. I will give you some pointers in the last section about how to move forward.

And you can download this handbook in PDF / ePub / Mobi format for free.

How much JavaScript you need to know to use React

Before jumping straight into React, you should have a good understanding of some core JavaScript concepts.

You don't have to be a JavaScript expert, but I think you need a good overview of:

- Variables
- Arrow functions
- Work with objects and arrays using Rest and Spread
- Object and array destructuring
- Template literals
- Callbacks
- ES Modules

How to install React

There are a few different ways to install React.

To start with, I highly recommend one approach, and that's using the offi
create-react-app.

create-react-app is a command line application, aimed at getting you up

You start by using **npx**, which is an easy way to download and execute Node.js commands without installing them.

See my npx guide here: https://flaviocopes.com/npx/

npx comes with **npm** (since version 5.2). If you don't have npm installed already, do it now from https://nodejs.org (npm is installed with Node).

If you are unsure which version of npm you have, run **npm -v** to check if you need to update.

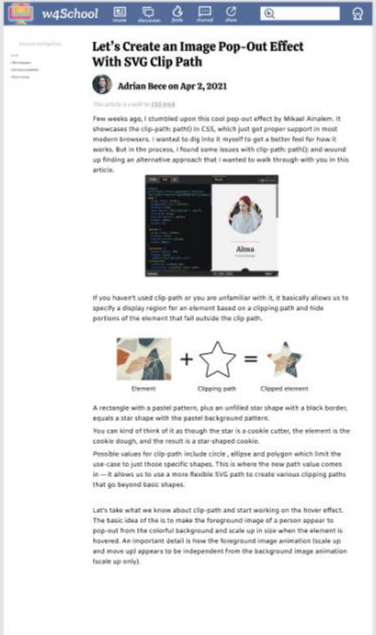
> Tip: check out my OSX terminal tutorial at https://flaviocopes.com/macos-terminal/ if you're unfamiliar with using the terminal. It applies to Mac and Linux.

When you run **npx create-react-app <app-name>**, **npx** is going to download the most recent **create-react-app** release, run it, and then remove it from your system.

This is great because you will never have an outdated version on your system, and every time you run it, you're getting the latest and greatest code available.

Credit to : https://www.freecodecamp.org/news/react-beginners-handbook/

CSS practice



w4School course discussion feeds channel share

Lesson navigation

Let's Create an Image Pop-Out Effect With SVG Clip Path

Adrian Bece on Apr 2, 2021

This article is credit to CSS trick

Few weeks ago, I stumbled upon this cool pop-out effect by Mikael Ainalem. It showcases the clip-path: path() in CSS, which just got proper support in most modern browsers. I wanted to dig into it myself to get a better feel for how it works. But in the process, I found some issues with clip-path: path(); and wound up finding an alternative approach that I wanted to walk through with you in this article.

If you haven't used clip-path or you are unfamiliar with it, it basically allows us to specify a display region for an element based on a clipping path and hide portions of the element that fall outside the clip path.

Element + Clipping path = Clipped element

A rectangle with a pastel pattern, plus an unfilled star shape with a black border, equals a star shape with the pastel background pattern.

You can kind of think of it as though the star is a cookie cutter, the element is the cookie dough, and the result is a star-shaped cookie.

Possible values for clip-path include circle , ellipse and polygon which limit the use-case to just those specific shapes. This is where the new path value comes in — it allows us to use a more flexible SVG path to create various clipping paths that go beyond basic shapes.

Let's take what we know about clip-path and start working on the hover effect. The basic idea of the is to make the foreground image of a person appear to pop-out from the colorful background and scale up in size when the element is hovered. An important detail is how the foreground image animation (scale up and move up) appears to be independent from the background image animation (scale up only).

Conclusion

Limitations

There are still some imperfections in the project. Since this project is implemented on Figma, some operations cannot be presented in the prototype, such as the functionality of the input, and auto-zoom adapting the screen size. And the layout of elements on each page should be slightly adjusted, especially the relationship of size and tint. Furthermore, users who have evaluated the prototype are almost with IT background and already learned a lot about programming. So, the evaluation cannot reflect the programming beginners. To improve, more iteration can be made, and the subjects should be more comprehensive.

Achievements with respect to project requirements

This project aims to develop a website that teaches university students the basics of web development. This stage is mainly for developing a usable system. Our website includes several contents essential to website development. For example, we design HTML and CSS courses and practices. We also include a web application framework, such as ReactJS. In addition, we think Python is important for today's website design, so, Python and Python bottle courses and practices are listed on our website. For a detailed list of all the contents used in our website, please refer to Appendix A.

We also implement core functions mentioned in the project brief. Our website allows users to register a new account using email. While later registered users can log in with their email and corresponding passwords. Users would have unique profiles after successful login, in which they are able to change some details about profiles (e.g., language set and word size). After that, users can view the knowledge base. For instance, users can access different courses and practices, including Python bottle, JavaScript, HTML, Python, ReactJS and CSS. In addition, we also implement the functions for website administrators. Apart from the normal functions, they can add, delete and mute users. Not only users can view different courses, but they can also message other users to communicate with them. Our website keeps track of the message history, which enables users to view the message history. A discussion forum also exists to let users and/or admin message each other.

Word count:3425

Appendix

Appendix A – content reference

The detailed contents of our website are listed below:

JavaScript course:

https://www.w3schools.com/js/DEFAULT.asp

Framework ReactJS:

https://www.freecodecamp.org/news/react-beginner-handbook/

CSS course:

https://www.freecodecamp.org/news/react-beginner-handbook/

CSS practice:

https://css-tricks.com/lets-create-an-image-pop-out-effect-with-svg-clip-path/

Python course:

https://realpython.com/invalid-syntax-python/

Python practice:

https://leetcode.com/problems/merge-k-sorted-lists/

HTML course:

https://www.w3schools.com/html/default.asp

HTML Practice:

https://www.w3schools.com/html/exercise.asp

Python bottle course:

https://bottlepy.org/docs/dev/index.html

Python bottle practice:

https://bottlepy.org/docs/dev/tutorial_app.html

Appendix B – PACT analysis

<u>People</u> use <u>technologies</u> to undertake <u>activities</u> in <u>contexts</u>

<u>People</u>

- Students from The University of Sydney
- Website admin(s)

<u>Technologies</u>

- Access to the Internet
- Input the keyword about the part of web development what they want
- View the tutorial material about what they want to learn
- Material came from the web-development-related courses from University
- Keyboard and mouse
- Tablet
- Smart phones

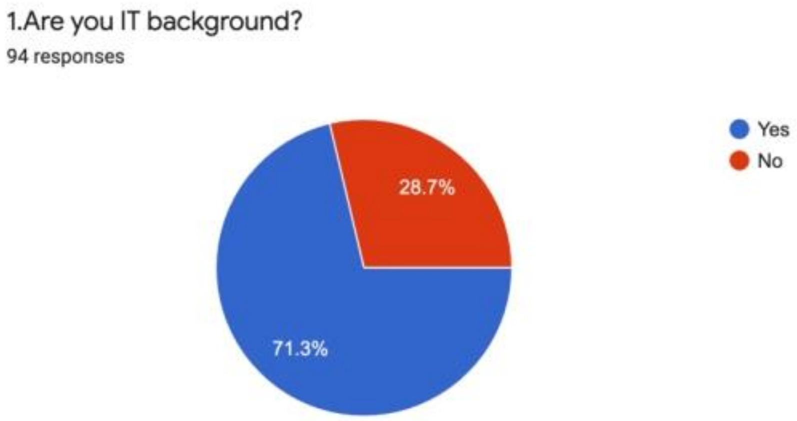
<u>Activities</u>

- Learn the basic skill of website development
- Discuss and solve problems with other developers

<u>Contexts</u>

- Related to web development skills and knowledge

Appendix C – survey result

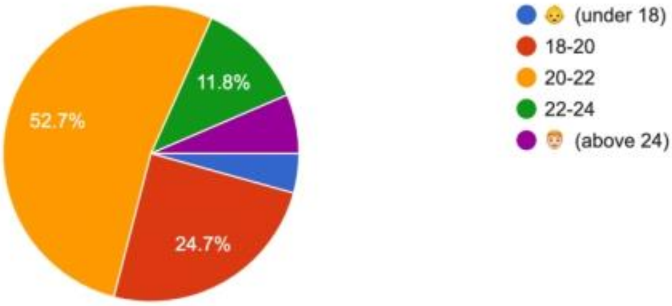


1.Are you IT background?

94 responses

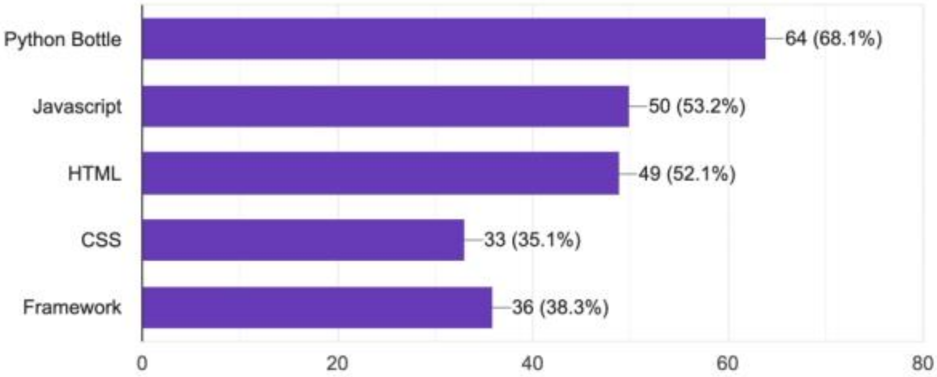
2.How old are you?

93 responses



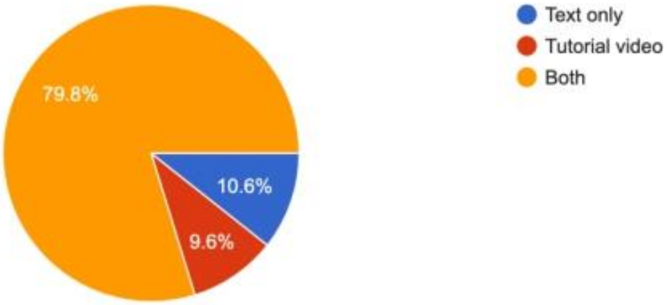
3.What do you expect from our website?

94 responses



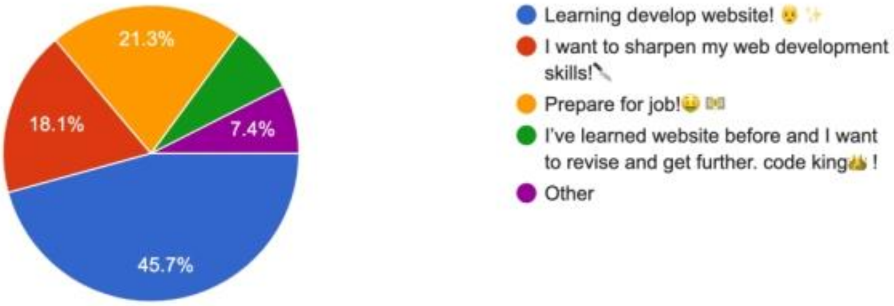
4.Which kind of learning method would you like to use?

94 responses



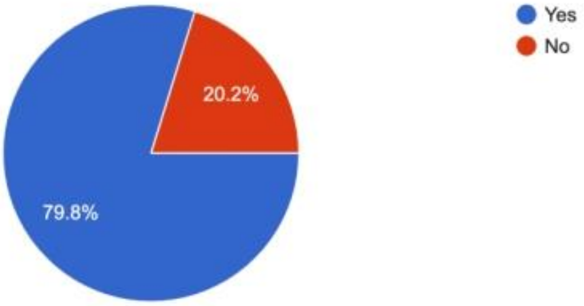
5.Your purpose of using our website

94 responses

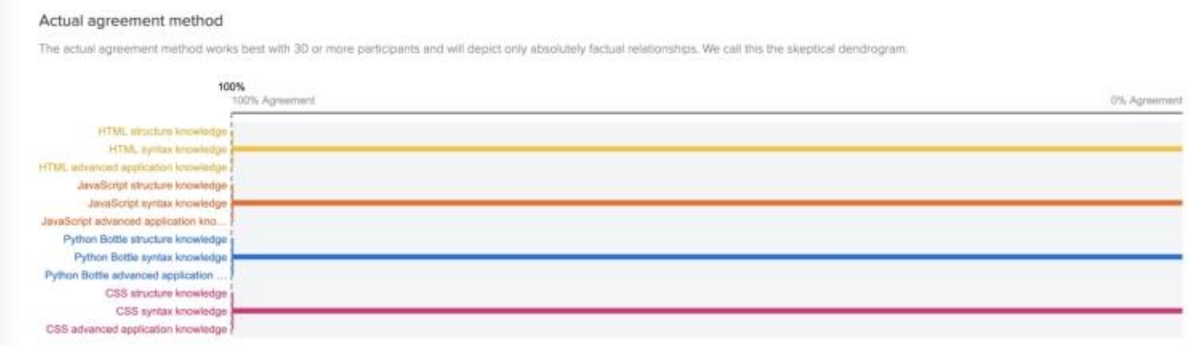


6.Do you need Multi-language support? (subtitle)

94 responses

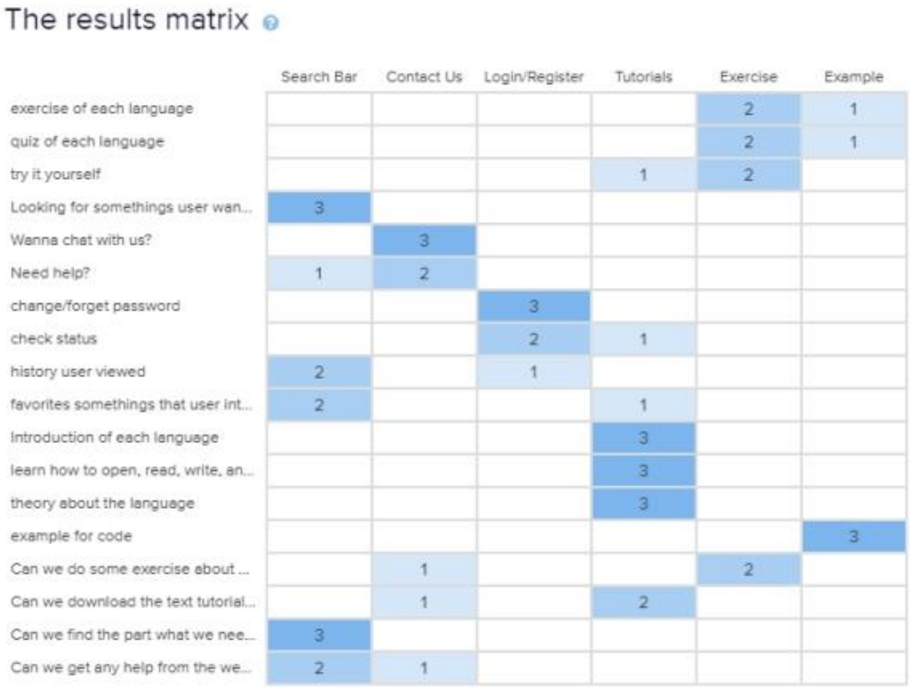


Appendix D – card sorting result



Actual agreement method

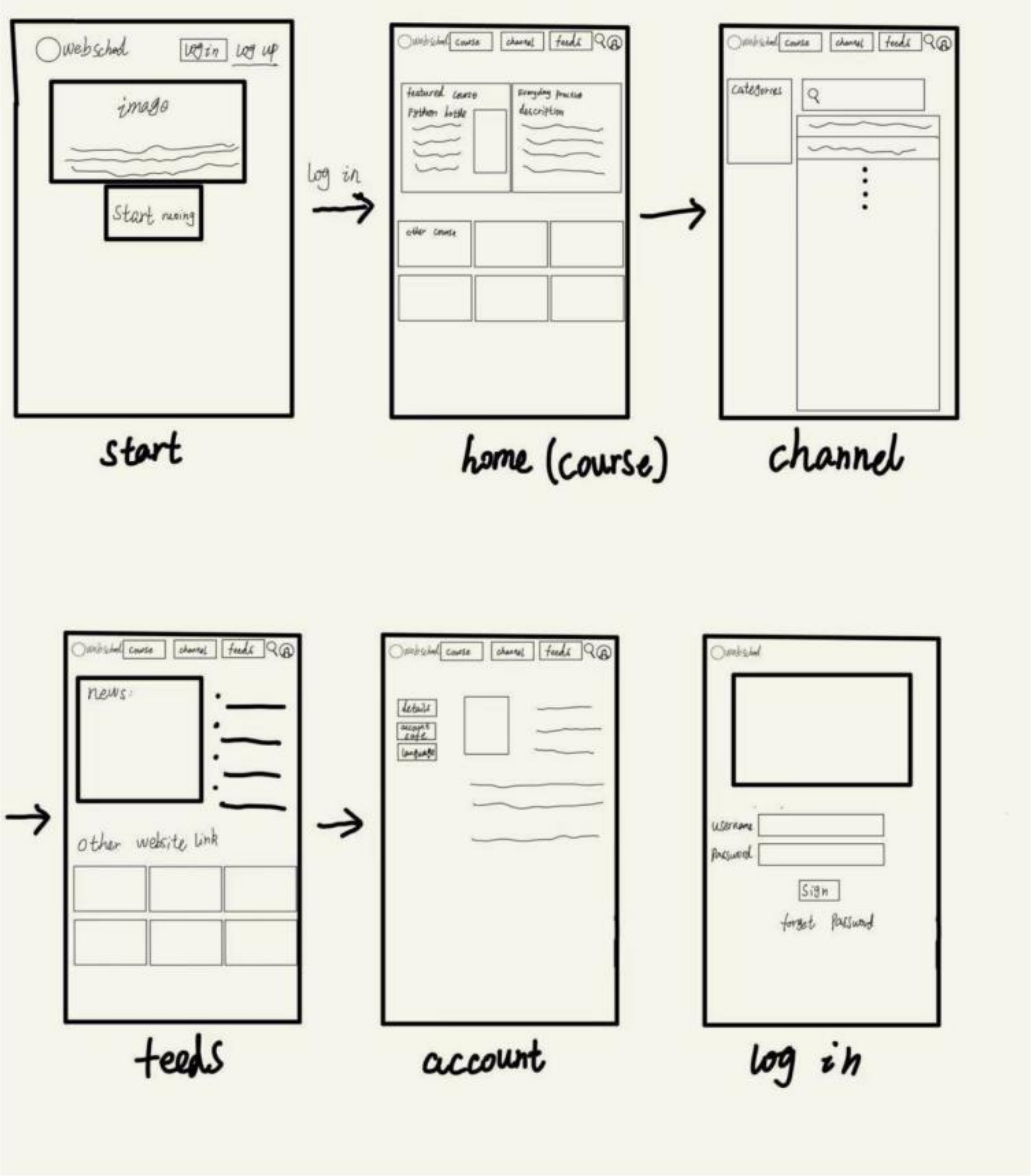
The actual agreement method works best with 30 or more participants and will depict only absolutely factual relationships. We call this the skeptical dendrogram.



The results matrix

	Search Bar	Contact Us	Login/Register	Tutorials	Exercise	Example
exercise of each language					2	1
quiz of each language					2	1
try it yourself				1	2	
Looking for somethings user wan...	3					
Wanna chat with us?		3				
Need help?	1	2				
change/forget password			3			
check status			2	1		
history user viewed	2		1			
favorites somethings that user int...	2			1		
Introduction of each language				3		
learn how to open, read, write, an...				3		
theory about the language				3		
example for code						3
Can we do some exercise about ...		1			2	
Can we download the text tutorial...		1		2		
Can we find the part what we nee...	3					
Can we get any help from the we...	2	1				

Appendix E – wireframe



Appendix F – First usability testing

User - Estelle Wang

Participant

Name: Estelle Wang

Age: 21

Degree: Bachelor of Advanced Computing

Gender: Female (She/her)

	Pages	**User Response (Any issues?)**	**Additional Comments**
1.	Sign up	None	
2.	Log in	None	Email login might be too long, we could use unikey
3.	View Bottle course	None	Bottle image could be smaller
4.	View JavaScript course	None	Could be set in higher resolution
5.	View HTML practice	None	
6.	View Python practice	None	

7.	Message a user	None	
8.	View message history	Task set inappropriately since there is no message listed	
9.	View Forum	None	
10.	Manage profile	None	
11.	View Feeds	None	

Interview Questions

1. What are your general opinions on the website design?

   Generally okay.

2. Which pages and specific elements do you think needs improvement?

   Other course in the course page is confusing, there are four links to one page. Featured courses other courses could be put in one page with featured courses prioritized.

3. Any features or functionalities you feel like is needed?

   Nothing else :)

User - Encheng Qu

Participant

Name: Encheng Qu

Age: 21

Degree: Bachelor of Software Engineering

Gender: Male (He/him)

	Pages	**User Response (Any issues?)**	**Additional Comments**
1.	Sign up	None	
2.	Log in	None	
3.	View Bottle course	None	The font so big, img should be small
4.	View JavaScript course	Should be add native	Could be set in higher resolution for pic.
5.	View HTML practice	None	There are only questions without any tutorials.
6.	View Python practice	None	
7.	Message a user	None	
8.	View message history	None	

9.	View Forum	None	Filter is not clear
10.	Manage profile	None	Fonts could be different
11.	View Feeds	None	Messy

Interview Questions

1. What are your general opinions on the website design?

   All good in structure and logic.

2. Which pages and specific elements do you think needs improvement?

   Course Page needed be improved, cannot find what I need in time and effectively.

3. Any features or functionalities you feel like is needed?
   No

User - Shiyuan Li

Participant

Name: Shiyuan Li

Age: 20

Degree: Bachelor of Advanced Computing

Gender: Male (he/him)

	Pages	**User Response (Any issues?)**	**Additional Comments**
1.	Sign up	No, pretty good	
2.	Log in	No problem	
3.	View Bottle course	Good	Bottle logo is too big
4.	View JavaScript course	The size of the font in the menu is too small	It will be better if the menu can be hided.
5.	View HTML practice	Good	
6.	View Python practice	The size of the font in the menu is too small	
7.	Message a user		Should improve this function and make DM achievable

8.	View message history		
9.	View Forum		
10.	Manage profile	Avatar take too many places in this page	
11.	View Feeds	No subscribe button of daily news.	

Interview Questions

1. What are your general opinions on the website design?

   Some defect exists, but mostly OK.

2. Which pages and specific elements do you think needs improvement?

   Other course, Quiz and Tutorial buttons all goes to the same page, which is very confused. They can be combined, which will save place for another potential functional button.

3. Any features or functionalities you feel like is needed?

   Subscribe button in Feeds.

User – Tony

Participant

Name: Tony

Age: 21

Degree: Undergraduate

Gender: male

	Pages	**User Response (Any issues?)**	**Additional Comments**
1.	Sign up	No, pretty good	
2.	Log in	No problem	
3.	View Bottle course	Good	
4.	View JavaScript course	Very good	
5.	View HTML practice	Good	
6.	View Python practice	Good	
7.	Message a user	Can't use the message since I can't type a word or DM with others and also cannot make a group talk.	Should improve this function and make DM achievable

8.	View message history	Can’t use this function since I can’t use message	
9.	View Forum	I can only see one forum in this website, and I can click on the others	
10.	Manage profile	I can’t manage my profile, but I can check my profile	Give the authorize to manage my own profile
11.	View Feeds	I can see the feeds, but I cannot click on them	Should be able to check multiple feeds

Interview Questions

1. What are your general opinions on the website design?

Pretty good about log in and create account and also seeing course but the other functions are seemed not prepared enough.

2. Which pages and specific elements do you think needs improvement?

Message manage profile and feeds.

3. Any features or functionalities you feel like is needed?

Nothing needed to be added but to improvement those functions I talked above.

User - Xiwen Zhang

Participant

Name: Xiwen Zhang

Age: 21

Degree: Bachelor of law

Gender: Female (She/her)

	Pages	User Response (Any issues?)	Additional Comments
1.	Sign up	None	
2.	Log in	None	Better change words "email" to "username"
3.	View discussion board	The participant chose the button channel	The image of button discussion and channel could be extremely similar
4.	View course	The size of the font in the menu is too small	The button of the course like a course schedule
5.	View HTML practice	None	There are only questions without any tutorials.
6.	View CSS course	The participant tries to find backward button	Needs to add a back button, otherwise users have to go back to course pages and choose another course then go to CSS course, which is too complex.

7.	Message a user	None	Same problem with q3
8.	View message history	None	
9.	Manage profile	None	None
10.	View Feeds	None	The button of feeds is too ambiguous

Interview Questions

1. What are your general opinions on the website design?

   Too many buttons are ambiguous

2. Which pages and specific elements do you think needs improvement?

   Website needs a back button

3. Any features or functionalities you feel like is needed?

   Aesthetics design should be upgraded

Appendix G – First usability testing

User – Venna

Participant

Name: Venna

Age:19

Degree: Bachelor of Engineering

Gender: Female (She/Her)

	Pages	**User Response (Any issues?)**	**Additional Comments**
1.	Sign up	None	
2.	Log in	None	
3.	View Bottle course	None	
4.	View JavaScript course	None	
5.	View HTML practice	None	‘Practice & course’ could be larger to draw more attention
6.	View Python practice	None	
7.	Message a user	None	Add friend button in channel page could be improved

8.	View message history	None	
9.	View Forum	None	
10.	Manage profile	None	Has a little issue on locating
11.	View Feeds	None	Need a record and load button to feel more natural

Interview Questions

1. What are your general opinions on the website design?
   - Search function could be only for course and practice & quiz
   - Go-back button in feeds
   - Adopted to Usyd canvas system, practice section could be embedded in related courses
   - The colour theme in the feeds could be improved.

2. Which pages and specific elements do you think needs improvement?

   Header in html course page has error (navigate to profile page)

3. Any features or functionalities you feel like is needed?
   Share function with website screenshot downloaded or Website bookmark

User - Christine (Yuwei)

Participant

Name: Christine (Yuwei)

Age:20

Degree: Bachelor of Science

Gender: Female (She/Her)

	Pages	User Response (Any issues?)	Additional Comments
1.	Sign up	None	It is good. But what does a resetting password need? Feel like only a security question is not enough….
2.	Log in	None	
3.	View Bottle course	None	I have looked at all the courses and practices. It is well formatted.
4.	View JavaScript course	None	
5.	View HTML practice	None	
6.	View Python practice	None	
7.	Message a user	None	Page 7 is so big that it cannot display on my side, even when I tried to zoom out.
8.	View message history	None	

9.	View Forum	None	
10.	Manage profile	None	
11.	View Feeds	None	

Interview Questions

1. What are your general opinions on the website design?

   I like the header, where I can (am oriented to?) jump to course, discussion, channel… very conveniently. The search bar also looks good, although I cannot use it now.

   It is also convenient that I can view my profile at the top right.

2. Which pages and specific elements do you think needs improvement?

   The total website is too full and a little bit messy. If there are some empty spaces, I personally will feel better. Also, I find the web page is not fixed, it can roll in horizontal direction. I will prefer a website with smaller font size and fixed horizontal direction. I notice that I can customise the font size by Setting. But It is a little bit weird.

   Moreover, I find that if I am redirected to other pages, there is no immediate way if I want to go back.

3. Any features or functionalities you feel like is needed?

   I think the functionalities are sufficient.

User - Irene Qu

Participant

Name: Irene Qu

Age: 21

Degree: Bachelor of Software Engineering

Gender: Male (He/him)

	Pages	User Response (Any issues?)	Additional Comments
1.	Sign up	None	
2.	Log in	None	
3.	View Bottle course	None	
4.	View JavaScript course	None	
5.	View HTML practice	None	
6.	View Python practice	None	
7.	Message a user	None	
8.	View message history	None	

9.	View Forum	None	Other thread cannot be clicked, and cannot reply the thread.
10.	Manage profile	None	
11.	View Feeds	None	

Interview Questions

1. What are your general opinions on the website design?

   All good in structure and logic.

2. Which pages and specific elements do you think needs improvement?

3. Any features or functionalities you feel like is needed?
   No

User - Sijia Yu

Participant

Name: Sijia Yu

Age: 22

Degree: Bachelor of Computer Science

Gender: Female

	Pages	**User Response (Any issues?)**	**Additional Comments**
1.	Sign up	Good	Font size seems too large
2.	Log in	Good	
3.	View Bottle course	Good	
4.	View JavaScript course	Good	
5.	View HTML practice	Good	
6.	View Python practice	Good	
7.	Message a user	Unable to type in anything yet	Users’ icon can be shown in the contact’s list.
8.	View message history	Good	

9.	View Forum	The layout is good, but not able to switch to other forum windows	The comment box is a bit small compare to the content part. Users may feel uncomfortable if they want to type in long contents.
10.	Manage profile	The ‘upload pic’ option is not available.	The ‘Moments’ section is unmeaningful, no explanation.
11.	View Feeds	Good	

Interview Questions

1. What are your general opinions on the website design?

   The font size of pages is not consistent, may cause users hard to read.

2. Which pages and specific elements do you think needs improvement?

   The contents/icons of course & quiz home page seems a bit overlapping. Maybe try to use only one icon for each programming language and give choices for course and quiz under each language.

   The icons of user can be added to the contacts list of chatting window, so that users can distinguish people easier.

3. Any features or functionalities you feel like is needed?
   The website can add some functions to allow users to share some pages/articles/courses that they feel useful or interesting to their friends.

User – Gaohao ming

Participant

Name: Gaohao ming

Age: 20

Degree: Bachelor of law

Gender: Female (She/Her)

	Pages	**User Response (Any issues?)**	**Additional Comments**
1	Sign up	None	
2	Log in	None	
3	View Bottle	None	
4	course View JavaScript course	None	Navigation have the same background colour with content
5	View HTML practice	None	
6	View Python practice	None	

7	Message a user	None	
8	View message history	None	
9	View Forum	None	
10	Manage profile	None	Safety should in the setting
11	View Feeds	None	

Interview Questions

1. What are your general opinions on the website design?

   All good in structure and logic.

2. Which pages and specific elements do you think needs improvement?

   Navigation have the same background colour with content

3. Any features or functionalities you feel like is needed?

   Some backgrounds colour should be change

Appendix H – PACT Analysis

People

- Students from The University of Sydney
- Website admin(s)

Technologies

- Access to the Internet
- Input the keyword about the part of web development what they want
- View the tutorial material about what they want to learn
- Material came from the web-development-related courses from University
- Keyboard and mouse
- Tablet
- Smartphones

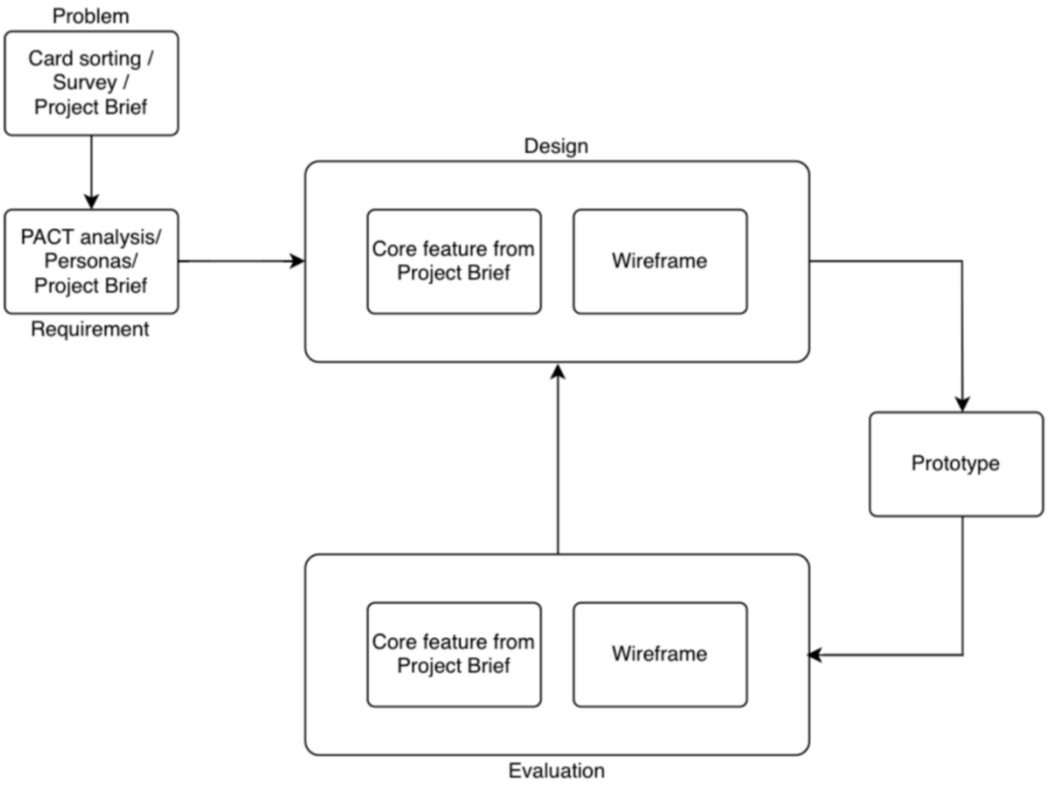
Activities

- Learn the basic skill of website development
- Discuss and solve problems with other developers

Contexts

- Related to web development skills and knowledge

Appendix I – Graphical representation of group's journey


Problem
Card sorting / Survey / Project Brief
PACT analysis/ Personas/ Project Brief
Requirement
Design
Core feature from Project Brief
Wireframe
Prototype
Core feature from Project Brief
Wireframe
Evaluation