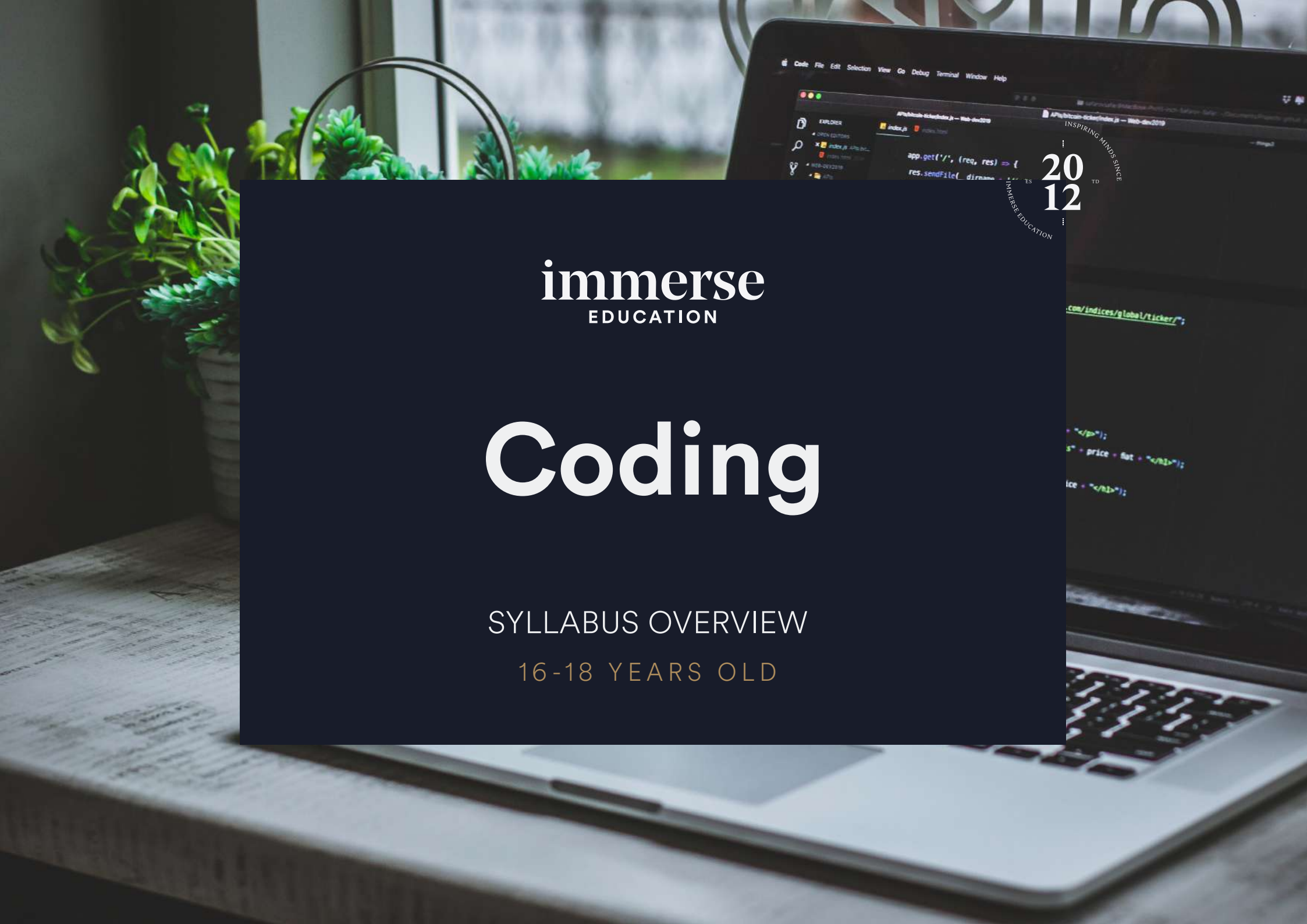




immerse
EDUCATION

Coding

SYLLABUS OVERVIEW

16-18 YEARS OLD

immerse
EDUCATION

About the Programme

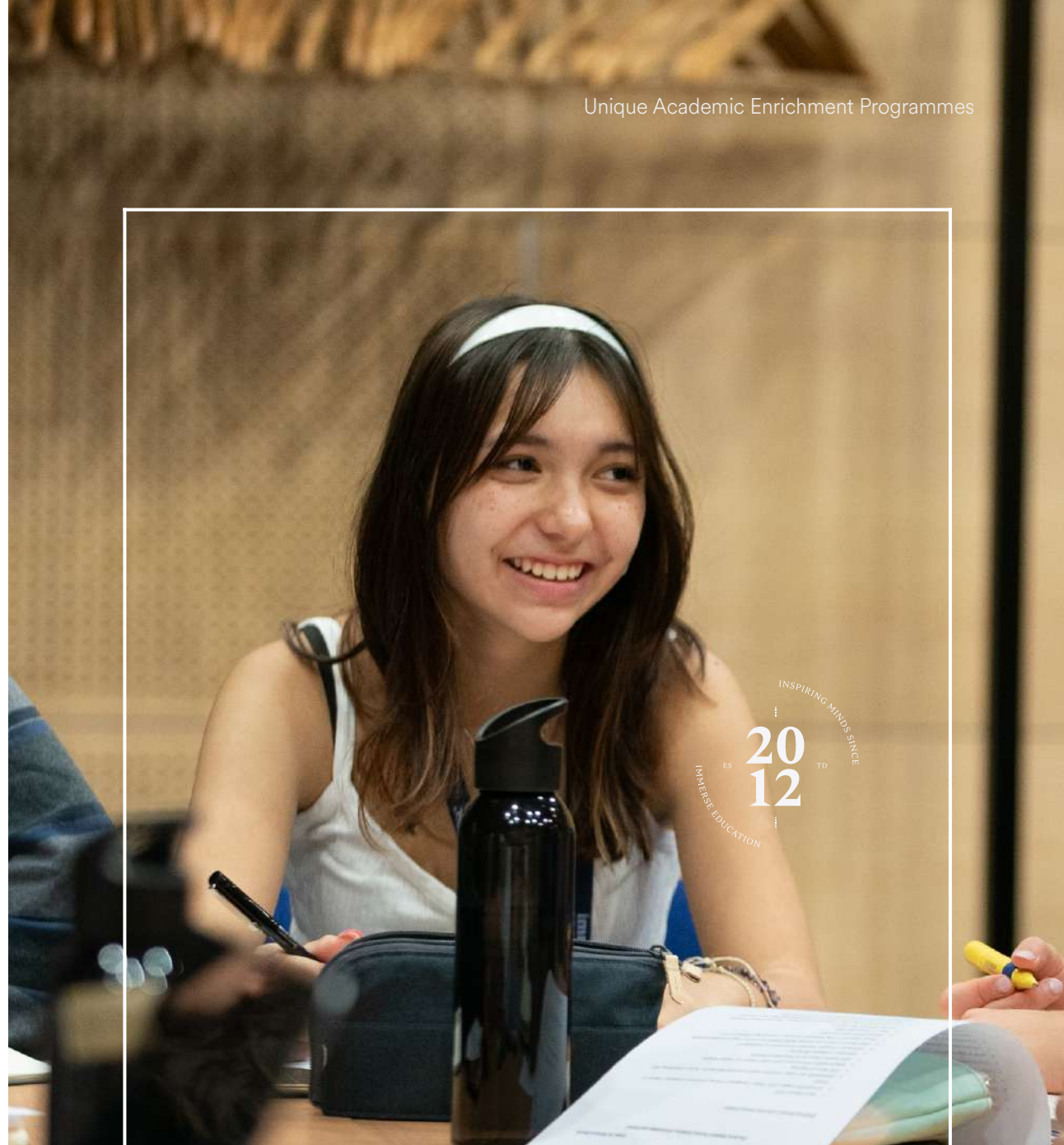
The aim of these introductory programmes is to provide participants with academically challenging content in a classroom environment based on the university style of learning. Through academic sessions, the programmes also offer young students unique and valuable insights into what it would be like to study their chosen subject at an advanced level.

This Syllabus Overview provides a summary of the topics and subject areas that participants can encounter during their studies with Immerse. It has been carefully created by our expert tutors who are members of faculty at world-leading universities, and who have experience in teaching undergraduate students.

Academic Sessions

The academic sessions at Immerse are arranged into modules to enable participants to explore a broad range of topics over the course of their programme. The modules included in this syllabus overview are indicative but not prescriptive.

Tutors are encouraged to include their own specialisms and also focus on any particular areas of interest expressed by participants within the class. They may choose to provide further detail on a specific topic, or they may include new material and information that builds on the knowledge already





Personal Project

Each programme includes an element of individual work, generally termed the 'Personal Project'. This can take many forms but is commonly an essay or presentation delivered on the final day of the programme. Participants will receive feedback on this work which may also be mentioned in the participant evaluation which is provided in writing by the tutor once the programmes have ended.



Preparatory work

Some tutors may ask participants to complete some preparatory work, such as reading or a series of exercises in advance of the programme. Participants are strongly encouraged to complete this work since it will be included in the opening sessions of the programme. Any preparatory tasks will be provided in advance of the programme directly to the participant.

Academic Difficulty

As all of our programmes are designed to provide a unique introduction to advanced material, the syllabus will be academically challenging at times.

This is something to be excited about and all of our tutors will encourage and support participants throughout the programme. Immerse Education aims to develop every participant regardless of ability, and our tutors will adapt their teaching to individual needs.

Aim of the Coding Programme

Being able to code is an important part of being a computer programmer. Creating and editing websites, applications and software, all require coding skills, one of the most sought-after skills needed across a vast majority of industries today. The coding programme is designed to teach you the basics of computer languages and interactions, from the foundations of coding to its complexities, and is ideal for participants interested in machine learning and technology. By the end of this programme, you will have a solid foundation of coding languages, know how to solve common coding problems, and be able to apply your newly built coding skills to a wide range of areas such as game design and website development.

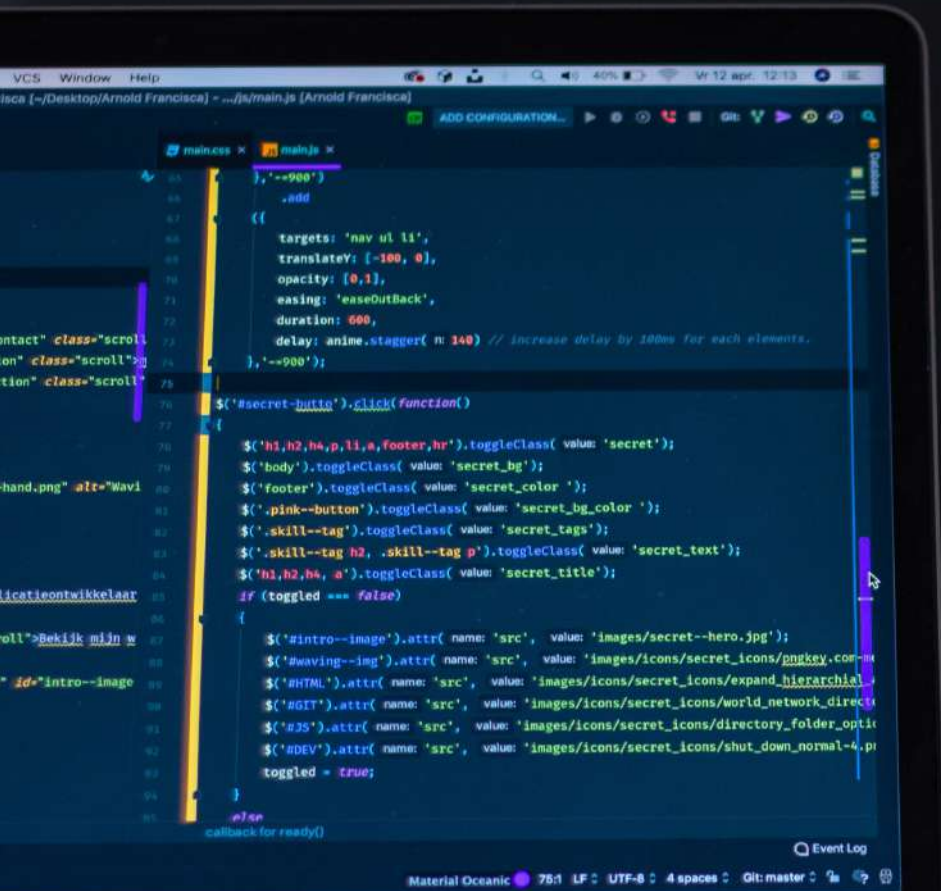
Introduction to Coding

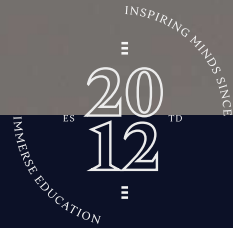
Coding is the process of telling a computer what to do. You provide the computer instructions via 'code', the computer's language, in order for it to perform a specific action. This can be as simple as adding code on a webpage to change the colour font to the code that informs traffic lights to change behaviour based on the flow of traffic. Coding is now integral to a vast number of industries, including healthcare, businesses, and schools (to name a few). In this module, you'll learn the basics of coding, establishing a foundation that will be built on throughout the rest of the course.



Introduction to Python

In this module, you'll explore the most widely used programming language: Python. You will learn how to interpret error messages as well as read basic Python syntax, all while forming an understanding of basic programming concepts. You will have the opportunity to apply your understanding of Python to the real world by creating your own simple app designed to solve a problem.





Introduction to Game Design

This module takes a look at coding through a game design lens. You will learn how to use code to custom behaviours and game mechanics using coding bestpractices. By the end of this module, you will have a firm understanding of essential programming concepts such as variables, branching, looping, events and how these tools are the building blocks of game design.

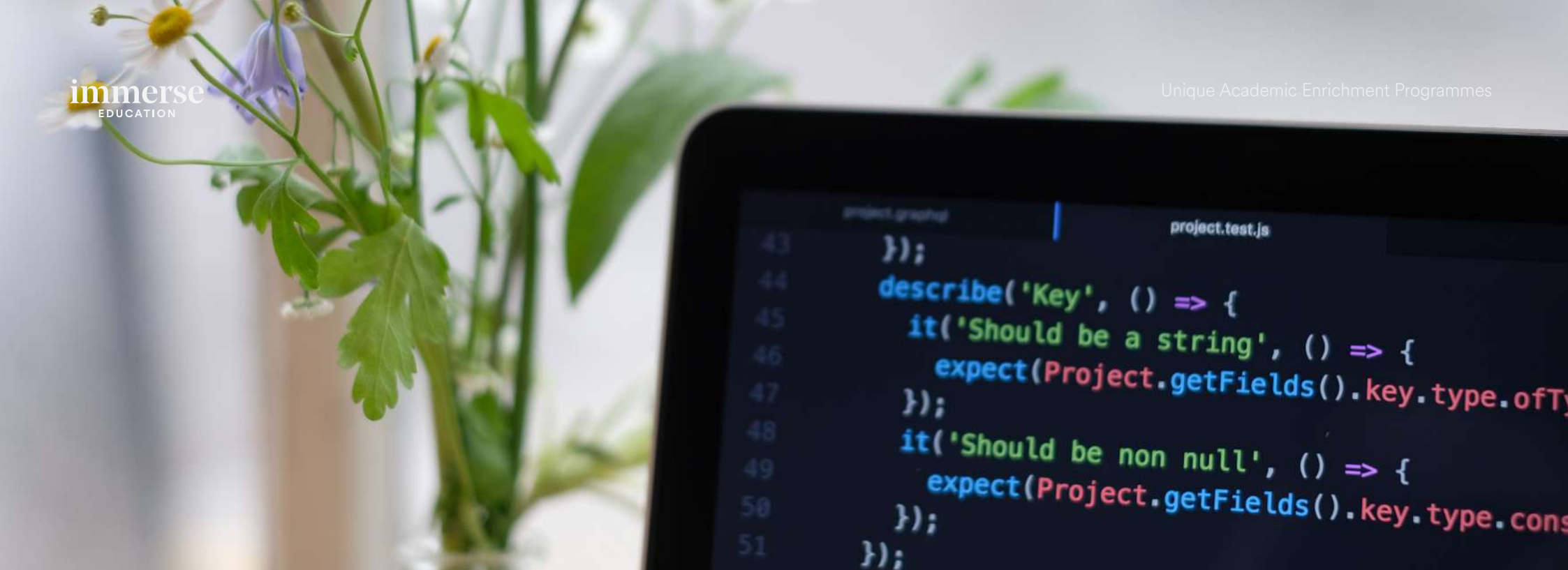
```

1  <script>
2    import sidebarController from './sidebarController.js'
3    import mementoes from '../store.js'
4    import './sidebar.scss';
5  </script>
6
7  <div class="sidebar">
8    <button class="add-memento" on:click=sidebarController.addMemento>Add Memento</button>
9
10   <ul class="mementoes">
11     {#each $mementoes as memento}
12       <li class="memento-item" class:active=memento.isActive>
13         {memento.title}
14       </li>
15     </ul>
16   </div>

```

Java Coder

In this unit, you will be introduced to Java Coder meanwhile learning how to theoretically examine coding logic using flowcharts and pseudo code. You will continue building on your understanding of coding best practices built in previous units as well as learning how to solve coding problems through solution building, as well as learning how to write code independently. You will also learn how to minimise the number of lines of code and functions in order to develop a more efficient coding process.



```
project.graphql | project.test.js
43   });
44   describe('Key', () => {
45     it('Should be a string', () => {
46       expect(Project.getFields().key.type.ofType).toBe(String);
47     });
48     it('Should be non null', () => {
49       expect(Project.getFields().key.type.constraints).toContain(NonNullType);
50     });
51   });
```

Coding Languages

In this unit, we will take a brief look at the various coding languages available to use in computer programming for different purposes. You will learn the differences between coding languages and develop an understanding of the languages that are easier and more convenient to use for different purposes. We will explore certain programming languages that are more machine-friendly, making them faster to execute and more memory-efficient, but harder for humans to understand.

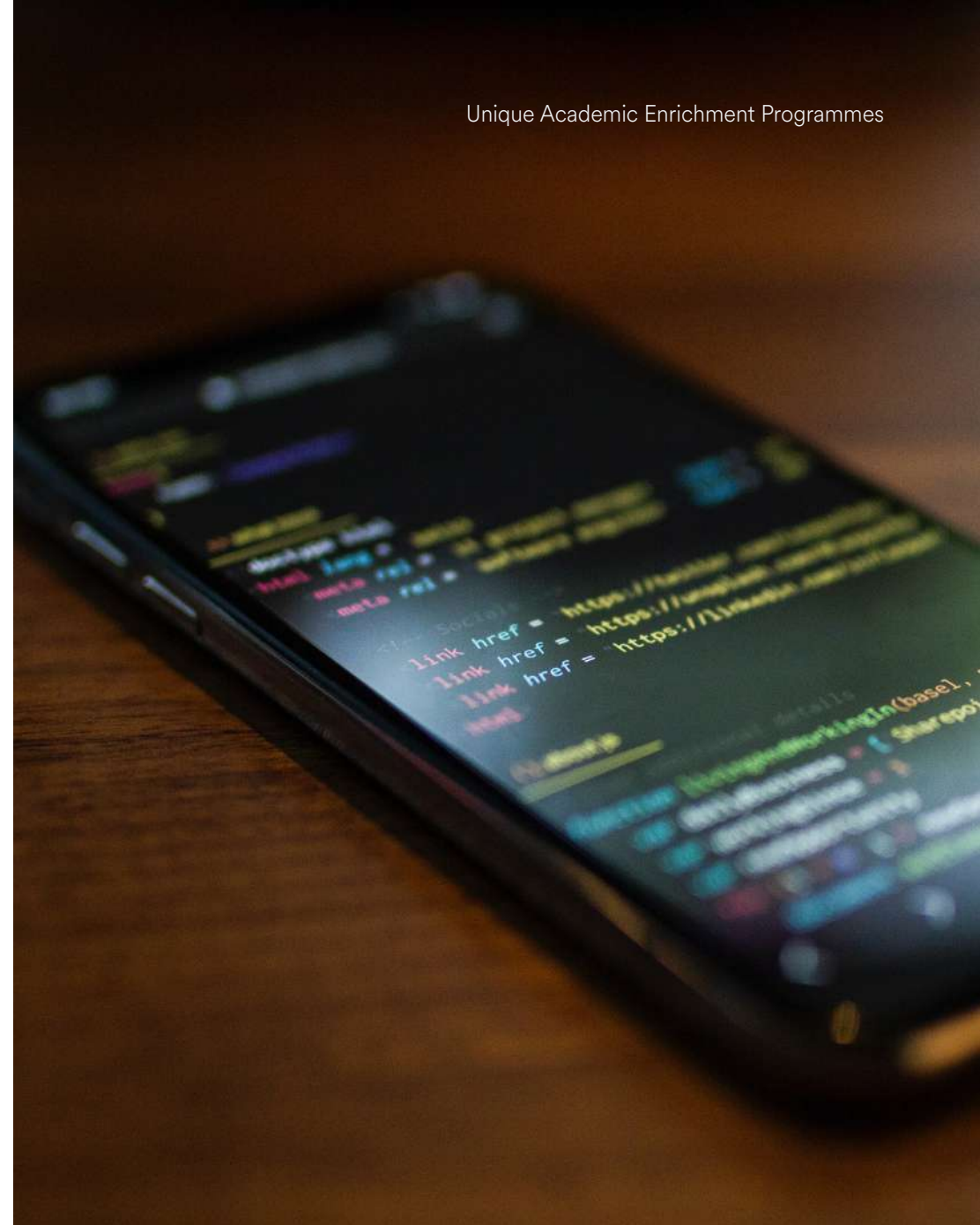


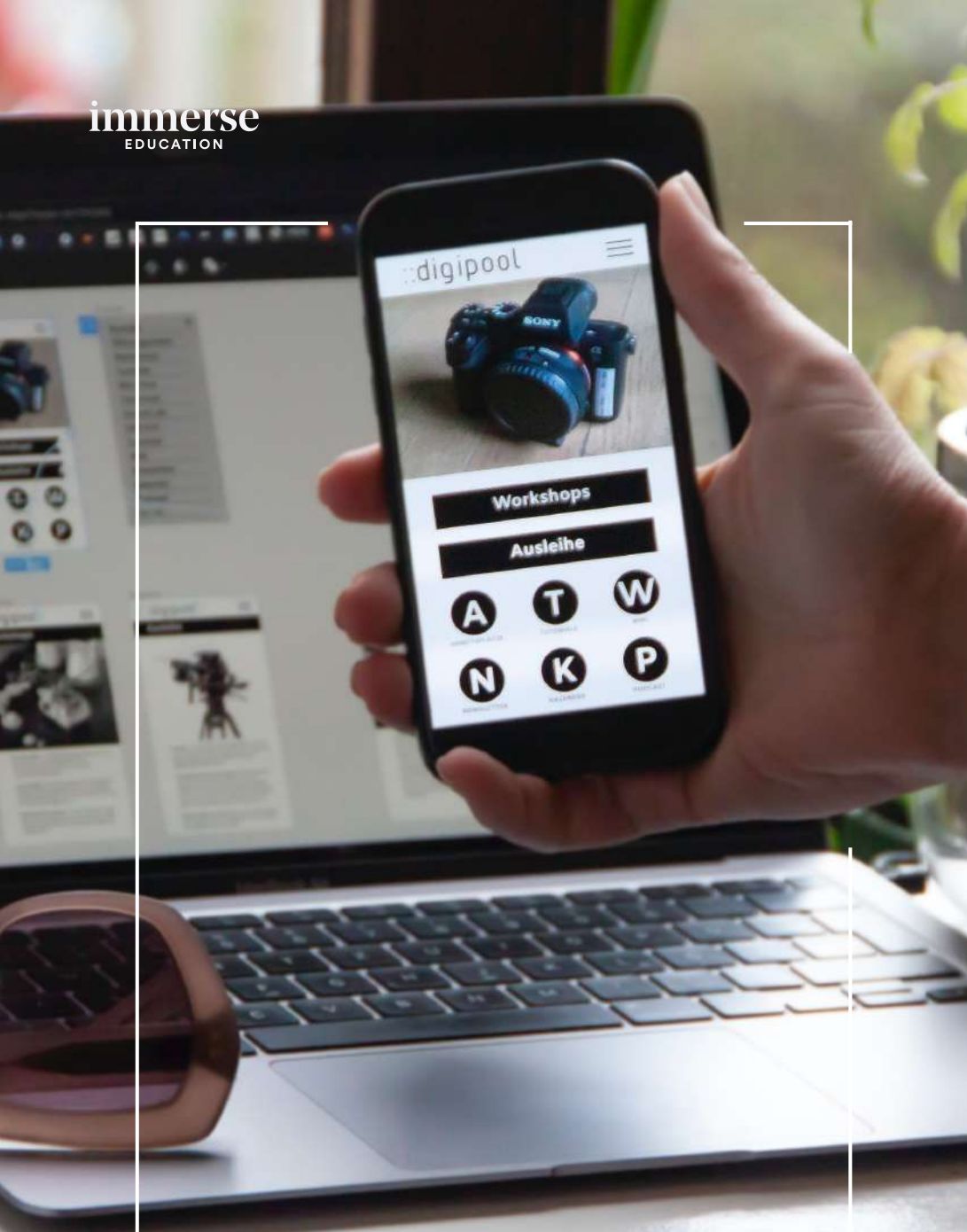
Web Development Masterclass

The growth of languages like JavaScript and frameworks like Angular, React, and Node JS have over the years made web development a lot easier and faster. In this unit, we'll take a look at how JavaScript and the relevant frameworks work together to provide the essential tools for web development and apply these to your own web building project.

Coding for Mobiles & Apps

Did you know that the average mobile phone in 2022 is faster than the average computer ten years ago? Coding has played a pivotal role in the way phones work, which is why in this module we'll take a look at how coding has changed over time in relation to mobile devices and the revolutions in mobile app development. You will apply your knowledge to create a more complex mobile app, using programming concepts to solve problems and build user-oriented solutions.





Design and Analysis of Algorithms

An algorithm is a sequence of steps used to solve a problem in computer programming. The design and analysis of algorithms are essential when designing processes to solve different types of problems in the branch of computer science and information technology. This module introduces the fundamental concepts of designing strategies, analysis of algorithms, as well as graph theory and sorting methods. By the end of this unit, you will also have a basic understanding of the Complexity Theory and its applications in the world of coding.

```

auth.service.ts | environment.prod.ts | environment.ts | config.xml | package-lock.json | db.service.ts | test.ts | polyfills.ts
1 | import { GooglePlus } from "@ionic-native/google-plus-ngx";
2 | import { Platform } from "@ionic/angular";
3 |
4 | import { LoadingController } from "@ionic/angular";
5 | import { Storage } from "@ionic/storage";
6 |
7 | @Injectable({
8 |   providedIn: "root"
9 | })
10 | export class AuthService {
11 |   users: Observable<any>;
12 |
13 | constructor(
14 |   private afAuth: AngularFireAuth,
15 |   private db: DbService,
16 |   private router: Router,
17 |   private gplus: GooglePlus,
18 |   private platform: Platform,
19 |   private loadingController: LoadingController,
20 |   private storage: Storage
21 | ) {
22 |   this.users = this.afAuth.authState.pipe(
23 |     switchMap(user => user ? db.doc<users>(user.uid) : of(null))
24 |   );
25 |
26 |   this.handleRedirect();
27 | }
28 |
29 | uid() {
30 |   return this.users
31 |     .pipe(
32 |       take(1),
33 |       map(u => u && u.uid)
34 |     )
35 |     .toPromise();
36 | }
37 |
38 | async anonymousLogin() {
39 |   const credential = await this.afAuth.auth.signInAnonymously();
40 |   return await this.updateUserData(credential.user);
41 | }
42 |
43 | }
44 |
45 |
46 |
47 |
48 |
49 |
50 |

```

OUR AWARDS AND ACCREDITATIONS



immerse
EDUCATION

Inspiring Great Young Minds

WWW.IMMERSE.EDUCATION