



SOFTWARE TESTING ACADEMY

Testowanie zwinne (Agile) Wprowadzenie *R2 – Materiały szkoleniowe*

Projekt The Software Testers Academy został zrealizowany przy wsparciu finansowym Unii Europejskiej. Niniejsza publikacja odzwierciedla jedynie stanowisko autora, a Komisja nie ponosi odpowiedzialności za ewentualne wykorzystanie zawartych w niej informacji.

Autorzy: Amaris Consulting – <https://www.amaris.com/>



Co-funded by
the European Union

Spis treści

0. WPROWADZENIE DO KURSU TESTOWANIA ZWINNEGO.....	3
1. STRUKTURA KURSU TESTOWANIA ZWINNEGO.....	4
2. TRENER TESTOWANIA ZWINNEGO.....	8
3. SPOŁECZNOŚĆ TESTERÓW.....	9
3.1. FORUM WEWNĘTRZNE.....	9
3.2. INNE SPOŁECZNOŚCI.....	9
4. NAJCZĘSCIEJ ZADAWANE PYTANIA.....	11

Licencja publiczna



Niniejsza publikacja © 2023 opracowana przez the Software Testing Consortium Partners jest na licencji Attribution-NonCommercial-NoDerivatives 4.0 International. Licencja dostępna pod adresem: <http://creativecommons.org/licenses/by-nc-nd/4.0/>

0. Wprowadzenie do kursu testowania zwinnego

Założeniem projektu Software Testing Academy jest utworzenie pierwszej tego typu akademii dla wszystkich, którzy chcą rozwijać swoją karierę w dziedzinie testowania oprogramowania. W tym celu, w ramach partnerstwa zostaną opracowane **trzy ścieżki edukacyjne: testowanie w Agile, testowanie automatyczne oraz testowanie dostępności**, wybrane w oparciu o obecne i pojawiające się potrzeby na rynku pracy.

Pierwszym obszarem, na którym skupimy się w ramach tego szkolenia, jest „**testowanie zwinne**”, którego głównym celem jest zapoznanie studentów z tematem i podstawowym sposobem myślenia, aby mogli polegać na swojej wiedzy i podjąć pierwsze (zaawansowane) kroki w testowaniu.



Familiarity

Can describe key aspects of a tester's everyday life and the tool kits.



Mindset

Internalize the mindset of questioning and trying everything & working agile.



Understanding

Can match test types with quality characteristics.



Doing

Can create test cases and bug reports.

Dzięki temu szkoleniu mamy nadzieję pomóc studentom wejść w nową rolę poprzez:

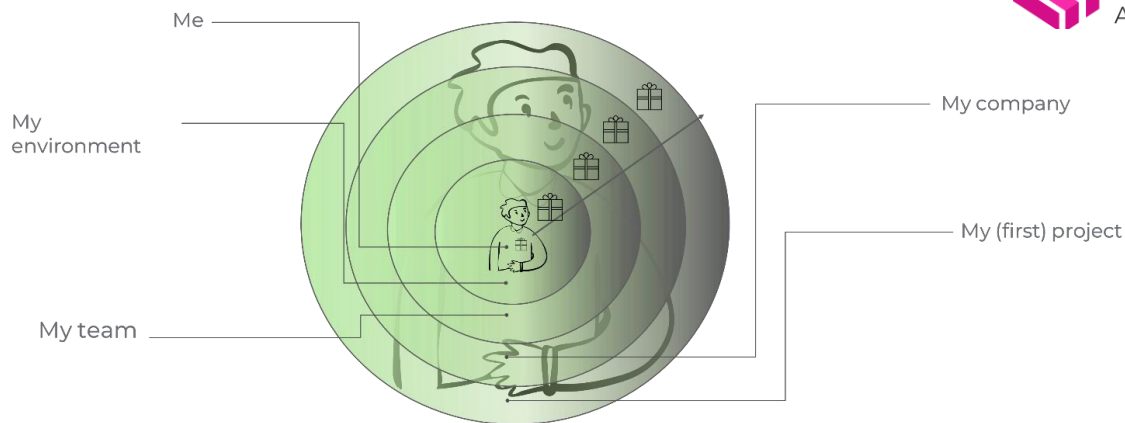
- Podkreślenie specyfiki testowania opartego na wnioskach i intuicyjności procedury.
- Zapoznanie z terminologią i koncepcją.

1. Struktura kursu testowania zwinnego

Przede wszystkim **nauczysz się funkcjonować jako niezależny tester**, co oznacza samodzielne testowanie (**Obszar „Ja”**). Następnie nauczysz się, jak testować w swoim bliskim otoczeniu (**Obszar „Moje otoczenie”**). Jest tyle ciekawych rzeczy, które możesz odkryć samodzielnie!

Dzięki obszarowi **„Ja”** dowiesz się o niektórych działaniach związanych z testowaniem, z których codziennie korzystasz, nie mając świadomości, iż są one procesem testowania. Na przykład ucząc się korzystać z telefonu komórkowego często myślisz – co się stanie, jeśli nacisnę ten przycisk albo wykonam tę czynność? To proces testowania, ale prawdopodobnie nie wiedziałeś o tym! Korzystając z pierwszego obszaru zapoznasz się z informacjami dotyczącymi podstawowych warunków przeprowadzania testów, a my pomożemy Ci zrozumieć, iż testy są obecne w naszym codziennym życiu, nawet jeśli są niewidoczne!

Następnie przejdziesz do kolejnego obszaru, tj. **„Moje otoczenie”**. Wiedząc, iż tak naprawdę wszyscy jesteśmy testerami, zaczniesz zauważać rzeczy wokół siebie z nastawieniem początkującego testera, próbując testować rzeczy, których wcześniej nie testowałeś, lub zaczniesz postrzegać aplikacje w swoim telefonie komórkowym w nieco inny sposób. Krok po kroku wejdziesz w nową rolę, pamiętając o procesach testowania, które od teraz zaczną kształtować twoją codzienną przyszłą pracę. **Proces uczenia się składa się zatem z pięciu etapów – nazywanych „obszarami” – które opierają się na sobie nawzajem** (zob. ilustracja poniżej).



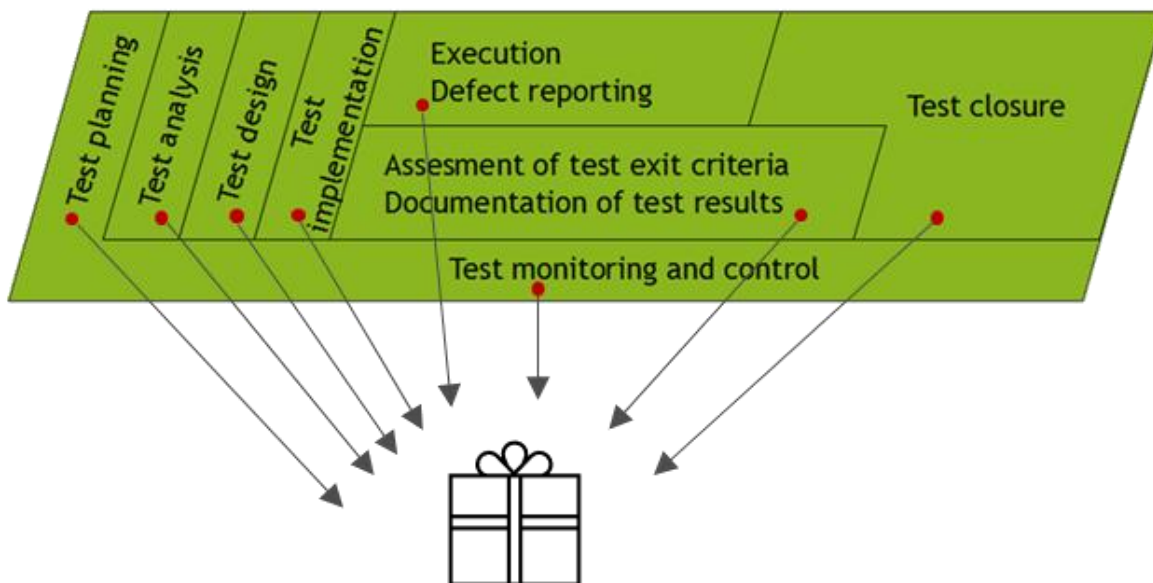
Struktura obszaru prezentowana podczas tego szkolenia umożliwi Ci identyfikację wad, nauczysz się, jak należy zgłaszać wady, napisać przypadek testowy lub zdobyć wiedzę na temat sposobu opracowania przypadku testowego w oparciu o wymogi aplikacji.

Trzeci obszar koncentruje się na uczeniu się testowania w zespole (**Obszar „Zespół”**), zatem na tym etapie dowiesz się, co jest potrzebne do współpracy z innymi w procesie testowania i opracowywania oprogramowania.

Czwarty obszar dotyczy sposobu, w jaki testujemy na poziomie przedsiębiorstwa (**Obszar „Moje przedsiębiorstwo”**).

Piąty, a zarazem ostatni obszar umożliwi Ci wykorzystanie w praktyce wszystkiego, czego dotychczas nauczyłeś się jako tester, abyś mógł doskonalić swoje umiejętności praktyczne (**Obszar „Mój pierwszy projekt”**) przy pomocy przykładowego projektu, który przygotowaliśmy dla Ciebie.

Poniższy wykres pokazuje, iż w każdym obszarze zamieściliśmy dla Ciebie niewielki „prezent”, co oznacza, iż każdy obszar zawiera sekcje związane z poszczególnymi fazami testów. Innymi słowy, **z każdego obszaru możesz dowiedzieć się trochę o każdym etapie procesu testowania**, a w miarę jak będziesz przechodzić do kolejnego obszaru, będziesz pogłębiać swoją wiedzę na temat tych etapów.



Szkolenie zostało zaplanowane **jako proces rozwoju, tak aby można było dowiedzieć się o procesie testowania w oparciu o pigułki wiedzy**, które zostały „zapakowane” z odpowiednimi materiałami na każdy z pięciu etapów kształcenia. Każdy z nich zawiera różnorodne materiały i ćwiczenia, umożliwiające podjęcie pierwszych kroków w testowaniu.

Każdy obszar podzielony jest na sekcje, które zawierają różne elementy:

- **Tematy:** Ogólna wizja lekcji.
- **Cele:** Opis celów edukacyjnych dla poszczególnych lekcji.
- **Wyniki:** Twoja wiedza i doświadczenie praktyczne zdobyte w trakcie lekcji.
- **Materiały:** Różne materiały, niezbędne do analizy w celu ukończenia lekcji, np. artykuły, nagrania wideo,...
- **Materiały dodatkowe:** Wykaz dodatkowych zasobów zalecanych do analizy w celu uzyskania pełniejszej wiedzy związanej z konkretnymi tematami istotnymi dla testerów.
- **Ćwiczenia:** Lista ćwiczeń, podczas których będziesz mógł wykorzystać wiedzę zdobytą w ramach lekcji.
- **Test:** Otrzymasz test składający się z kilku pytań służących do dokonania samooceny swojej wiedzy. Uzyskanie pozytywnego wyniku z testów jest niezbędne do otrzymania certyfikatu.

Podczas kursu spróbuj opanować jak najwięcej terminów i nabyć wiedzę na ich temat, by móc stosować je w życiu codziennym jako przyszły tester ds. oceny jakości. Zachowaj materiały i analizuj je tyle razy, ile uważasz za konieczne do opanowania ich treści. Wykonaj ćwiczenia jak najlepiej, aby móc w pełni przećwiczyć nabytą wiedzę.



Studenci, którzy pomyślnie ukończą szkolenie, mają większe szanse na dostanie pracy – pamiętaj, że osoby, które osiągną najlepsze wyniki zostaną wymienione i zaproponowane jako potencjalni kandydaci dla naszych partnerów branżowych!
Nie przegap szansy na rozpoczęcie kariery jako tester oprogramowania!

2. Trener prowadzący szkolenie zwinne

Jako student masz możliwość skontaktowania się ze **trenerem prowadzącym szkolenie zwinne**, który odpowie na Twoje pytania i udzieli wskazówek, w najlepszy sposób możliwy sposób, na temat wyszukiwania potrzebnych informacji lub planowania swojego procesu edukacji aby jak najlepiej wykorzystać kurs.

Więcej informacji na temat **roli trenera testowania zwinnego** znajduje się na naszych wpisach na blogu na poniższej stronie:



AGILE, ERASMUSPLUS, SOFTWARETESTINGACADEMY, TRAINING

Agile learning coaching: how-to keep students motivated

Do you want to know how we keep our learners motivated during their training? The solution is simple, thanks to our partner QualityMinds: agile learning coaching gives more motivation to learn! It has been proven that both personalised learning processes and coaching of participants lead to higher motivation and learning success (Schuster, 2018; Bloom, 1984)...



AGILE, ERASMUSPLUS, SOFTWARETESTING,
SOFTWARETESTINGACADEMY, TRAINING

Agile teaching: what is it?



AGILE, ERASMUSPLUS, SOFTWARETESTINGACADEMY,
TRAINING, WORKATESTER

Agile learning modules

Kliknij na 'Skontaktuj się z nami', aby skontaktować się z trenerem.

3. Społeczność testerów

3.1. Forum wewnętrzne

Dysponujemy wewnętrznym forum, na którym można podzielić się wątpliwościami, pomagać innym uczniom lub dyskutować o pewnych aspektach treści kursu. Aby uzyskać dostęp do społeczności wewnętrznej, kliknij [tutaj](#). **[Więcej informacji wkrótce!]**

3.2. Inne społeczności

Oto niektóre z najważniejszych forów testujących oprogramowanie, dyskusje i tablice ogłoszeń, do których możesz dołączyć i wykorzystać w nauce:

- **forum UTest:** Przyjazne dla użytkownika i aktywne forum dla testerów oprogramowania na wszystkich poziomach. Możesz zadawać pytania i odpowiadać na nie, dzielić się swoimi doświadczeniami oraz uczestniczyć w różnych projektach testowych i wyzwaniach.
- **Forum TESTHuddle:** Europejskie forum specjalistów ds. testowania oprogramowania. Możesz uzyskać dostęp do innowacyjnych treści, sieci kontaktów z rówieśnikami i dyskutować o wszelkich aspektach testowania oprogramowania.
- **Forum ds. badań jakości:** Jedno z najbardziej renomowanych i najstarszych forów w dziedzinie testowania oprogramowania. Można tutaj znaleźć prawie wszystko związane z testowaniem oprogramowania, od instruktażu, narzędzi, wskazówek, po pracę i wydarzenia.
- **Ministry of Testing Club :** Zawodowa społeczność testerów oprogramowania o zasięgu globalnym. Możesz dołączyć do klubu, aby współtworzyć lepsze testy, uzyskać dostęp do wyjątkowych zasobów i uczestniczyć w wydarzeniach online i offline.
- **[Grupy LinkedIn]:** Miejsce, w którym możesz połączyć się z innymi testerami oprogramowania na LinkedIn, największej na świecie sieci zawodowej. Możesz dołączyć do różnych grup związanych z testowaniem oprogramowania, takich jak:

- ✓ **Grupa ds. testów oprogramowania i zapewniania jakości**, licząca ponad 200,000 członków, zajmująca się takimi tematami jak metody testowania oprogramowania, narzędzia, techniki, najlepsze praktyki, tendencje, itp.
- ✓ **Ministry of Testing - The online Software Testing Community**: Społeczność licząca ponad 75,000 członków i specjalistów wysokiej jakości.
- ✓ **Testowanie oprogramowania i ocena jakości**: Grupa specjalistów ds. testów oprogramowania i zapewniania jakości, współpracujących ze sobą, pomagających znaleźć pracę, omawiać tendencje i problemy oraz tworzyć sieć kontaktów.
- ✓ **Testowanie i automatyzacja oprogramowania**: Grupa składająca się z ponad 350,000 członków-specjalistów ds. oprogramowania, którzy zamieszczają artykuły, zadają pytania i dyskutują na temat inżynierii jakości, testowania oprogramowania, automatyzacji, narzędzi testowych, wydajności, bezpieczeństwa, opracowywania oprogramowania, oceny jakości, Devops, testowania zwinnego i powiązanych tematów.
- **[Zapewnienie jakości oprogramowania SQA]**: Sieć licząca ponad 17.00 członków i ekspertów, którzy chcą dzielić się swoją wiedzą i poszerzać wiedzę na temat zapewniania jakości oprogramowania.
- **[Quora]**: Platforma, na której można zadawać wszelkie pytania i uzyskać odpowiedzi od ekspertów i innych użytkowników. Umożliwia śledzenie tematów związanych z testowaniem oprogramowania, takich jak techniki testowania oprogramowania, narzędzia do testowania oprogramowania, zapewnianie jakości oprogramowania, itp.
- **[Sieć SQA Stack Exchange]**: Sieć obejmuje 183 społeczności, wymieniających się informacjami w tym Stack Overflow, największej, najbardziej zaufanej społeczności internetowej umożliwiającej twórcom uczenie się, dzielenie się wiedzą i budowanie kariery zawodowej.

4. FAQ

Materiały dydaktyczne: Czy muszę zapoznać się ze wszystkimi tekstami i materiałami dydaktycznymi dotyczącymi zadania?

To zależy. Jeśli posiadasz już wiedzę na ten temat, możesz bezpośrednio przejść do ćwiczeń. Dodatkowo, na końcu kursu czasem znajdziesz dodatkowe ćwiczenia mające na celu poszerzenie swojej wiedzy.

Czas/długość: Ile czasu jest przewidziane na ukończenie kursu?

Bez obaw-z uwagi na indywidualne tempo nauki, możesz potrzebować więcej lub mniej czasu na wybrany temat.

Egzamin: Czy ukończenie kursu wiąże się z przystąpieniem do egzaminu?

Kurs nie kończy się egzaminem w rozumieniu szkolnym. Na końcu każdego obszaru znajdują się testy, które pomogą Ci sprawdzić swoją wiedzę. Do quizu możesz przystępować dowolną ilość razy, gdyż spełnia on jedynie funkcję narzędzia usprawniającego Twój proces uczenia się. Niemniej jednak, aby uzyskać świadectwo ukończenia kursu, konieczne jest pozytywne zaliczenie tych krótkich form sprawdzania wiedzy.

Ćwiczenia: Gdzie należy zachować wyniki ćwiczeń?

Możesz skorzystać z własnego notebooka, aby wykonać ćwiczenia. Jeśli chcesz, możesz podzielić się swoimi spostrzeżeniami na forum pod adresem <https://www.softwaretestingacademy.eu/>

Certyfikacja: Czy po zakończeniu szkolenia uzyskam certyfikat?

Tak, po uzyskaniu pozytywnych wyników z testów po każdym obszarze otrzymasz świadectwo ukończenia kursu przy co najmniej 65 % poprawnych odpowiedzi. Będziesz miał 3 próby na wykonanie testu!

Języki: Nie czuję się zbyt komfortowo pracując/ucząc się w języku angielskim. Czy istnieje możliwość nauki w innych językach?

Tak, obecnie szkolenie jest dostępne w pięciu językach: angielskim, hiszpańskim, portugalskim, niemieckim i polskim. Zapoznaj się z naszymi wytycznymi dotyczącymi nauki w innych językach: tutaj [\[Do opracowania -](#)

[link do wytycznych wyjaśniających stosowanie wtyczki do zewnętrznych stron internetowych](#)].



SOFTWARE TESTING ACADEMY

Testowanie zwinne

Część 1: Obszar „Ja”

R2 – Materiały szkoleniowe

Projekt The Software Testing Academy został zrealizowany przy wsparciu finansowym Unii Europejskiej. Niniejsza publikacja odzwierciedla jedynie stanowisko autora, a Komisja nie ponosi odpowiedzialności za ewentualne wykorzystanie zawartych w niej informacji.

Autorzy: Amaris Consulting – <https://www.amaris.com/>



Co-funded by
the European Union

Spis treści

SPIS TREŚCI	2
0. OBSZAR „JA”	4
0.1. CZEGO DOTYCZY TEN OBSZAR?	4
0.2. WYKORZYSTYWANIE WCZEŚNIEJSZEJ WIEDZY PRZED ROZPOCZĘCIEM TESTOWANIA	4
1. PLANOWANIE TESTÓW	6
1.1. TEMATY	6
1.2. CELE	6
1.3. WYNIKI	6
1.4. MATERIAŁY	6
1.5. MATERIAŁY DODATKOWE	7
1.6. ĆWICZENIA	7
2. ANALIZA TESTÓW	8
2.1. TEMATY	8
2.2. CELE	8
2.3. WYNIKI	8
2.4. MATERIAŁY	8
2.5. MATERIAŁY DODATKOWE	9
2.6. ĆWICZENIA	9
3. PROJEKTOWANIE TESTÓW	10
3.1. TEMATY	10
3.2. CELE	10
3.3. WYNIKI	10
3.4. MATERIAŁY	10
3.5. ĆWICZENIA	11
4. IMPLEMENTACJA TESTÓW	12
4.1. TEMATY	12
4.2. CELE	12
4.3. WYNIKI	12
4.4. MATERIAŁY	12
4.5. MATERIAŁY DODATKOWE	12
4.6. ĆWICZENIA	13
5. WPROWADZENIE DO AUTOMATYZACJI TESTÓW:	14
5.1. TEMATY	14
5.2. CELE	14
5.3. WYNIKI	14

5.4. MATERIAŁY	14
5.5. MATERIAŁY DODATKOWE	15
5.6. ĆWICZENIA.....	15
6. WYKONANIE TESTÓW: SAMOORGANIZACJA.....	16
6.1. TEMATY	16
6.2. CELE	16
6.3. WYNIKI.....	16
6.4. MATERIAŁY	16
6.5. MATERIAŁY DODATKOWE	17
6.6. ĆWICZENIA	17
7. ZAMKNIĘCIE TESTÓW	19
7.1. TEMATY	19
7.2. CELE	19
7.3. WYNIKI.....	19
7.4. MATERIAŁY	19
7.5. MATERIAŁY DODATKOWE	19
7.6. ĆWICZENIA	20
8. MONITOROWANIE I KONTROLA TESTÓW – ZARZĄDZANIE.....	21
8.1. TEMATY	21
8.2. CELE	21
8.3. WYNIKI.....	21
8.4. MATERIAŁY	21
8.5. MATERIAŁY DODATKOWE	22
8.6. ĆWICZENIA	22

Licencja publiczna



Niniejsza publikacja © 2023 opracowana przez the Software Testing Consortium Partners jest na licencji Attribution-NonCommercial-NoDerivatives 4.0 International. Licencja dostępna pod adresem:

<http://creativecommons.org/licenses/by-nc-nd/4.0/>

0. Obszar „Ja”

0.1. Czego dotyczy ten obszar?

Zacznijmy od ścieżki edukacyjnej „Testowanie zwinne”! Jak wiesz, będziesz stopniowo rozwijać się w ramach **swojej nowej roli jako tester zwinny**, pamiętając jednocześnie o procedurach testowania, definiujących codzienne podejście do pracy. Proces uczenia się dzieli się na pięć szczegółowych, odrębnych etapów, zwanych „obszarami”.



W obszarze „Ja” nabędziesz podstawowe umiejętności pozwalające na niezależną ocenę różnych elementów i opanowanie podstawowych koncepcji testowania często stosowanych w ramach rutynowych czynności.



Być może nie jesteś świadomy, iż w życiu codziennym przeprowadzasz testy na otaczających nas przedmiotach. Na przykład, gdy korzystasz z urządzenia mobilnego; instynktownie chcesz wiedzieć: „Co się stanie, jeśli aktywuję ten przycisk lub wykonam tę czynność?” To jest właśnie przykład testu, który często przechodzi niezauważenie.

W związku z powyższym, na tym wstępnym etapie zapoznasz z podstawową terminologią związaną z testowaniem i łatwiej zrozumiesz, iż testy są w znacznym stopniu zintegrowane z naszym codziennym życiem.

0.2. Wykorzystywanie wcześniejszej wiedzy przed rozpoczęciem testowania

Zanim przejdziemy do konkretnych pojęć, **sprawdźmy, co wiesz na temat testowania!** Zakładamy, iż przy omawianiu kolejnych tematów będziesz miał świadomość, w jaki sposób poprawi się Twoja wiedza dzięki temu szkoleniu.

Spróbuj odpowiedzieć na następujące pytania. Nie martw się, jeśli nie znasz odpowiedzi na wszystkie pytania, gdyż celem tego ćwiczenia jest jedynie próba spisania tego, co już wiesz na ten temat i dokonanie ponownej analizy pod koniec tego obszaru.



Testowanie zwinne – obszar „Ja”

- Czy masz jakąś wiedzę na temat testowania?
- Czy wiesz, jaki jest cel testowania?
- Czy wiesz, jaka jest różnica między testowaniem a jakością?

Zacznijmy zatem od pierwszego tematu w obszarze „Ja”!



1. Planowanie testów

1.1. Tematy



- Definicje testów i terminów z nimi związanych
- Poziomy testów

1.2. Cele

Jako przyszły tester będziesz w stanie **podać podstawowe definicje związane z testowaniem oraz poziomy testów.**



1.3. Wyniki



Po zakończeniu tego działu będziesz mógł:

- Zdefiniować pojęcie testu za pomocą własnych słów
- Wyjaśnić różnicę między weryfikacją a walidacją.
- Wymienić cztery główne poziomy testów i wyjaśnić każdy z nich.

1.4. Materiały

- Czym jest jakość? Definicja jakości, znaczenie jakości (przeczytaj do „Zapewnienie jakości”):

<https://techqualitypedia.com/quality/> (tekst)



- Zapewnianie jakości, kontrola jakości i testowanie:

<https://www.altexsoft.com/whitepapers/quality-assurance-quality-control-and-testing-the-basics-of-software-quality-management/> (tekst)

- Weryfikacja i walidacja:

<https://www.guru99.com/verification-v-s-validation-in-a-software-testing.html> (tekst)

1.5. Materiały dodatkowe

- Poziomy testów:

<https://artoftesting.com/levels-of-software-testing> (Tekst +
video: 4,5 min.)

- Weryfikacja a walidacja:

<https://www.youtube.com/watch?v=Zks3AectneU> (video, 6 min.)

- Testowanie oprogramowania, definicje:

<https://www.ibm.com/topics/software-testing> (tekst)

- Siedem zasad testowania:

<https://www.guru99.com/software-testing-seven-principles.html> (tekst)

1.6. Ćwiczenia

Po zapoznaniu się z materiałami spróbuj wykonać kilka ćwiczeń, aby wykorzystać nabytą wiedzę w praktyce.



Ćwiczenie 1:

Jak „oficjalnie” zdefiniować jakość? Dokonaj podsumowania w maksymalnie trzech zdaniach i porównaj je ze swoją własną definicją. Uwzględnij następujące kwestie: W jakim stopniu Twoja definicja była zbliżona do definicji zawartej w źródle? Czy według ciebie brakuje jakichkolwiek aspektów w oficjalnych definicjach?

Ćwiczenie 2:

Czy wiesz, jaka jest różnica między weryfikacją a walidacją? Podaj krótkie wyjaśnienie.

Ćwiczenie 3:

Czy możesz podać cztery główne poziomy testów wymienione w materiale? Wymień je i podaj krótkie wyjaśnienie dla każdego z nich.

Zadanie 4:

Czy możesz podać przykłady, w jakich sytuacjach należy zastosować każdy poziom testu?

Ćwiczenie 5:

Wybierz jedną ze swoich ulubionych stron internetowych i podaj przykład tego, co mógłbyś przetestować na każdym poziomie testowym.

2. Analiza testów

2.1. Tematy



- Wymogi i specyfikacje.

2.2. Cele

Jako przyszły tester będziesz w stanie zdefiniować termin „wymagania” i wyjaśnić ich znaczenie w opracowywaniu testów zgodnie z potrzebami klienta.



2.3. Wyniki



Po zakończeniu tego działu będziesz mógł:

- Wyjaśnić pojęcie „wymóg”.
- Przedstawić co najmniej dwa argumenty ukazujące jak ważna jest inżynieria oprogramowania.
- Podać różnicę między „wymogiem” a „specyfikacją”.

2.4. Materiały

- Definicja i pojęcia:

<https://www.altexsoft.com/blog/business/functional-and-non-functional-requirements-specification-and-types/> (tekst)

- Określenie wymagań i specyfikacji:

<https://argondigital.com/blog/product-management/requirements-vs-specifications-create-a-shared-vocabulary/> (tekst)

- Czym jest „wymóg” w inżynierii oprogramowania?

<https://techwithmaddy.com/what-is-a-requirement-in-software-engineering> (tekst)



2.5. Materiały dodatkowe

- Testowanie: cele i klasyfikacje:

<https://www.youtube.com/watch?v=sxoXCyEn8jY> (video, 12 min.)



- Wymagania, specyfikacje i różnice:

<https://wellfire.co/learn/requirements-and-specifications/> (tekst)

2.6. Ćwiczenia

Po zapoznaniu się z materiałami spróbuj wykonać kilka ćwiczeń, aby wykorzystać nabytą wiedzę w praktyce.



Ćwiczenie 1:

Spróbuj opracować własną definicję testu. Twoja definicja powinna być zapisana w pełnych zdaniach i nie powinna być dłuższa niż jedna czwarta strony. Spróbuj być przekonującym w swojej definicji.

Ćwiczenie 2:

Zdefiniuj znaczenie słowa „wymóg”. Zastanów się nad swoją ulubioną stroną internetową i wybierz jedną funkcję (na przykład login). Czy możesz sporządzić listę głównych wymogów, jakich oczekujesz w związku z tą funkcją?

3. Projektowanie testów

3.1. Tematy



- Projektowanie testów

3.2. Cele

Jako przyszły tester będziesz w stanie wyjaśnić, dlaczego projektowanie testu ma znaczenie jeśli chodzi o motywację, oraz będziesz mógł zaprojektować odpowiednie i innowacyjne testy.



3.3. Wyniki



Po zakończeniu tego działu będziesz mógł:

- Wyjaśnić jak ważny jest proces projektowania testu.

3.4. Materiały

- Projektowanie testu i definicje automatycznego projektowania testu:

https://en.wikipedia.org/wiki/Test_design (tekst)

- Czym jest projektowanie testu? Jak stworzyć projekt testu?

<http://tryqa.com/what-is-test-design-when-to-create-it/> (tekst)

- Znaczenie projektowania testu: proszę *przeczytać rozdział pt. „Znaczenie projektowania testu”*

<https://testautomationresources.com/software-testing-basics/software-test-design-techniques/> (tekst)



3.5. Ćwiczenia



Po zapoznaniu się z materiałami spróbuj wykonać kilka ćwiczeń, aby wykorzystać nabytą wiedzę w praktyce.

Ćwiczenie 1:

Spróbuj wyjaśnić za pomocą własnych słów, w jaki sposób wymogi i projektowanie testu współdziałają ze sobą.

Ćwiczenie 2:

Podaj co najmniej dwa powody uzasadniające znaczenie inżynierii wymogów.

4. Implementacja testów

4.1. Tematy



- Testowanie funkcjonalne
- Testowanie niefunkcjonalne

4.2. Cele

Jako przyszły tester będziesz w stanie opisać różnicę między testami funkcjonalnymi i niefunkcjonalnymi, aby prawidłowo przypisać cechy jakościowe, a tym samym usprawnić codzienną pracę.



4.3. Wyniki



Po zakończeniu tego działu będziesz w stanie:

- Zdefiniować własnymi słowami testy funkcjonalne i niefunkcjonalne.
- Podać różnicę między testami funkcjonalnymi i niefunkcjonalnymi.
- Przyporządkować testy funkcjonalne i/lub niefunkcjonalne do wymagań.

4.4 Materiały

- Różnice pomiędzy testowaniem funkcjonalnym i niefunkcjonalnym:

<https://www.guru99.com/functional-testing-vs-non-functional-testing.html> (tekst)



4.5. Materiały dodatkowe



- Testowanie funkcjonalne i niefunkcjonalne:
https://www.youtube.com/watch?v=ZUn35hxq_NM (video, 7 min.)

4.6. Ćwiczenia



Po zapoznaniu się z materiałami spróbuj wykonać kilka ćwiczeń, aby wykorzystać nabytą wiedzę w praktyce.

Ćwiczenie 1:

Czy możesz podać krótką definicję testów funkcjonalnych i нефункциональных oraz wskazać różnicę pomiędzy nimi? Spróbuj ją zapisać w maksymalnie 10 zdaniach.

Ćwiczenie 2:

Czy możesz podać przykłady dla każdego rodzaju testów (funkcjonalnych i нефункциональных)?

Ćwiczenie 3:

Wybierz jedną ze swoich ulubionych stron internetowych. Czy możesz pomyśleć o 5 różnych testach нефункциональных, które mógłbyś przeprowadzić? W jaki sposób, twoim zdaniem, testy te mogą przyczynić się do poprawy jakości produktu?

5. Wprowadzenie do automatyzacji testów:

5.1. Tematy



- Automatyzacja testów
- ✓ Koncepcja
- ✓ Definicje
- ✓ Ryzyko

5.2. Cele

Jako przyszły tester będziesz w stanie wyjaśnić podstawy automatyzacji testów, aby podążać zgodnie z trendami w testowaniu.



5.3. Wyniki



Po zakończeniu tego działu będziesz mógł:

- Wyjaśnić termin „automatyzacja testów”.
- Wymienić trzy zalety i trzy rodzaje zagrożeń związane z automatyzacją testów.

5.4. Materiały

- Automatyzacja testów, definicje:

<https://www.testim.io/blog/what-is-test-automation/> (tekst)

- Instruktaż do automatyzacji testów:

<https://www.guru99.com/automation-testing.html> (Tekst, w tym wideo, 7 min.)

- Korzyści i zagrożenia wynikające z automatyzacji testów:

<https://www.h2kinfosys.com/blog/benefits-risks-automation-testing/> (tekst)



5.5. Materiały dodatkowe



- Zalety i wady zautomatyzowanego testowania:
<https://youtu.be/UN-N7mIKbrY> (video 7 min.)

5.6. Ćwiczenia

Po zapoznaniu się z materiałami spróbuj wykonać kilka ćwiczeń, aby wykorzystać nabytą wiedzę w praktyce.



Ćwiczenie 1:

Proszę wyjaśnić własnymi słowami termin „automatyzacja testów” w maksymalnie pięciu zdaniach.

Ćwiczenie 2:

Po uzyskaniu wiedzy na temat testowania manualnego i zautomatyzowanego odpowiedz na następujące pytania: Jakie mogą być korzyści z automatyzacji testów? Jakie są potencjalne zagrożenia? Wymień trzy zalety i trzy rodzaje ryzyka.

6. Wykonanie testów: Samoorganizacja

6.1. Tematy



- Sposób myślenia testera oprogramowania.
Konkretnie:
 - Rola testera
 - Zarządzanie, w tym samoorganizacja.
- Inteligencja emocjonalna

6.2. Cele

Jako przyszły tester będziesz w stanie opisać sposób myślenia, proces dziedziczenia testów oraz włączanie ich do swojej codziennej pracy. Będziesz również mógł opisać za pomocą własnych słów czym jest „inteligencja emocjonalna”.



6.3. Wyniki



Po zakończeniu tego działu będziesz mógł:

- Wyjaśnić rolę testera w zespole opracowującym testy.
- Wymienić umiejętności, jakie powinien posiadać dobry tester, i określić, jakie umiejętności sam posiadasz i które z nich chcesz udoskonalić.
- Podać metody samoorganizacji.
- Zdefiniować termin „inteligencja emocjonalna” za pomocą własnych słów.

6.4. Materiały

- Typy testerów:

<https://thesocialtester.co.uk/tester-types-book/>

(Książka w formacie PDF do pobrania)



- Umiejętności niezbędne do testowania, narzędzia i obowiązki w zespole testującym:

<https://www.altexsoft.com/blog/engineering/qa-engineering-roles-skills-tools-and-responsibilities-within-a-testing-team/> (tekst)

- Sposób myślenia związany z testowaniem i definicje. Odpowiedzi uzyskane od społeczności testującej:

<https://club.ministryoftesting.com/t/what-does-testing-mindset-mean-to-you-whats-your-tester-mindset/26422> (tekst)

- Inteligencja emocjonalna: Wprowadzenie

<https://www.helpguide.org/articles/mental-health/emotional-intelligence-eq.htm> (tekst)

- Inteligencja emocjonalna w przywództwie:

https://www.mindtools.com/pages/article/newLDR_45.htm (tekst)

- Zarządzanie czasem: Techniki i narzędzia:

<https://www.usa.edu/blog/time-management-techniques/> (tekst)



6.5. Materiały dodatkowe

- Czy chcesz dowiedzieć się więcej o swojej inteligencji emocjonalnej? Wypełnij poniższy test:

<https://globalleadershipfoundation.com/geit/eitest.html> (tekst)

6.6. Ćwiczenia

Po zapoznaniu się z materiałami spróbuj wykonać kilka ćwiczeń, aby wykorzystać nabytą wiedzę w praktyce.



Ćwiczenie 1:

Z jakim rodzajem testera się identyfikujesz? Pamiętaj, że w większości przypadków wybierane jest połączenie kilku typów.

Ćwiczenie 2:

Jakie umiejętności potrzebne do przeprowadzania testów już posiadasz? Które z nich chcesz udoskonalić?

Ćwiczenie 3:

Jedną z niezbędnych umiejętności do zostania testerem oprogramowania, jest „Inteligencja emocjonalna”. Jak po przeczytaniu tej kwestii sformułowałbyś przy pomocy własnych słów ‘inteligencję emocjonalną? Jak ważna jest, w twojej ocenie, inteligencja emocjonalna w twojej pracy? Zapisz swoje przemyślenia maksymalnie w pięciu do dziesięciu zdaniach.

Zadanie 4:

Na tym etapie chcielibyśmy przetestować metodę samoorganizacji podczas nauki nowych treści. Z materiałów „Zarządzanie czasem” wybierz jedną technikę (taką, której nigdy wcześniej nie używałeś) i opisz, dlaczego ją wybrałeś i w jaki sposób, masz nadzieję, wpłynie ona na poprawę twojej pracy. Czego oczekujesz od tej techniki? Podaj co najmniej 5 punktów.

Ważne: Zachowaj swoją odpowiedź, ponieważ wykorzystasz ją do testowania kolejnych zadań.

7. Zamknięcie testów

7.1. Tematy



- Strategie zarządzania czasem

7.2. Cele

Dzięki podejściu retrospektywnemu będziesz w stanie zastanowić się nad swoimi wynikami, „punktem początkowym” i nową wiedzą zdobytą dzięki poprzednim lekcjom.



7.3. Wyniki



Po zakończeniu tego działu będziesz mógł:

- Wybrać nabytą metodę samoorganizacji i zastosować ją do danego zadania.

7.4. Materiały

- Strategie samoorganizacji i zarządzania czasem:
<https://www.planview.com/resources/articles/what-is-self-organizing-team/> (tekst)



- Techniki i narzędzia zarządzania czasem:
<https://www.usa.edu/blog/time-management-techniques> (tekst)

7.5. Materiały dodatkowe



- Strategie organizacji testujących:
<https://www.stickyminds.com/article/test-organization-strategies> (tekst)

7.6.Ćwiczenia



Po zapoznaniu się z materiałami spróbuj wykonać kilka ćwiczeń, aby wykorzystać nabytą wiedzę w praktyce. W ramach ćwiczeń do tych lekcji będziesz pracować nad ulepszeniem odpowiedzi podanych w ćwiczeniu 4 w lekcji 6. Samoorganizacja.

Ćwiczenie 1:

Jaką metodę samoorganizacji wybrałeś w poprzedniej lekcji? Czy uważasz ją za słuszną? Czy wybrałbyś ją ponownie, biorąc pod uwagę nabytą wiedzę po zakończeniu tej lekcji? Czy dzięki tej metodzie twoje oczekiwania zostały spełnione?

8. Monitorowanie i kontrola testów – zarządzanie

8.1. Tematy



- Zarządzanie testami i najlepsze praktyki.

8.2. Cele

Jako przyszły tester możesz wyjaśnić proces zarządzania testami i jego etapy, aby zawsze mieć lepszy obraz podczas pracy nad jednym etapem procesu testowania.



8.3. Wyniki



Po zakończeniu tego działu będziesz mógł:

- Wymenić zalety zarządzania testami.
- Podać etapy zarządzania testami i opisać każdy z nich w krótkim zdaniu.

8.4. Materiały

- Zarządzanie testami: Działania podejmowane w procesie testowania, oprogramowanie testowe i role testów (Przeczytaj 'Proces testowania', s. 18–23):

https://istqb-main-web-prod.s3.amazonaws.com/media/documents/ISTQB_CTFL_Syllabus-v4.0.pdf (do ściągnięcia pdf)

- Proces zarządzania testami w ramach testowania oprogramowania: <https://www.guru99.com/test-management-phases-a-complete-guide-for-testing-project.html> (tekst)
- Kierownik ds. testów, definicje, role, obowiązki: <https://www.freelancermap.com/blog/what-does-a-test-manager-do> (tekst)





8.5. Materiały dodatkowe

- Czym jest zarządzanie testami:
<https://www.youtube.com/watch?v=huv8zOGZ85E> (video, 6 min.)



8.6. Ćwiczenia

Po zapoznaniu się z materiałami spróbuj wykonać kilka ćwiczeń, aby wykorzystać nabytą wiedzę w praktyce.

(Należy zauważyć, że różni autorzy podsumowują i opisują poszczególne etapy w różny sposób. Bądź świadomy tych różnic. Polegamy na definicji ISTQB, ale widzimy także podobieństwo do innych definicji.)

Ćwiczenie 1:

Na czym polega zarządzanie testami?

Ćwiczenie 2:

Dlaczego uważasz, że zarządzanie testami jest ważne?

Ćwiczenie 3:

Wykaz etapów zarządzania testami:

- Planowanie
- Wykonanie



SOFTWARE TESTING ACADEMY

Testowanie zwinne (Agile)

Część 2: Obszar „Moje otoczenie”

R2 – Materiały szkoleniowe

Projekt The Software Testing Academy został zrealizowany przy wsparciu finansowym Unii Europejskiej. Niniejsza publikacja odzwierciedla jedynie stanowisko autora, a Komisja nie ponosi odpowiedzialności za ewentualne wykorzystanie zawartych w niej informacji.

Autorzy: Amaris Consulting – <https://www.amaris.com/>



Co-funded by
the European Union

Spis treści

0. OBSZAR „MOJE OTOCZENIE”	4
1. PLANOWANIE TESTÓW	5
1.1. TEMATY	5
1.2. CELE	5
1.3. WYNIKI.....	5
1.4. MATERIAŁY	5
1.5. MATERIAŁY DODATKOWE	6
1.6. ĆWICZENIA	6
2. ANALIZA TESTÓW	7
2.1. TEMATY	7
2.2. CELE	7
2.3. WYNIKI.....	7
2.4. MATERIAŁY	7
2.5. MATERIAŁY DODATKOWE.....	8
2.6. ĆWICZENIA	8
3. PROJEKTOWANIE TESTÓW	10
3.1. TEMATY	10
3.2. CELE	10
3.3. WYNIKI.....	10
3.4. MATERIAŁY	10
3.5. ĆWICZENIA.....	11
4. IMPLEMENTACJA TESTÓW	12
4.1. TEMATY.....	12
4.2. CELE.....	12
4.3. WYNIKI	12
4.4. MATERIAŁY.....	12
4.5. MATERIAŁY DODATKOWE	12
4.6. ĆWICZENIA.....	13
5. WYKONANIE TESTÓW	14
5.1. TEMATY	14
5.2. CELE	14
5.3. WYNIKI.....	14
5.4. MATERIAŁY	14
5.5. MATERIAŁY DODATKOWE	14
5.6. ĆWICZENIA	15
6. ZAMKNIĘCIE TESTÓW	16

6.1. TEMATY	16
6.2. CELE	16
6.3. WYNIKI	16
6.4. MATERIAŁY	16
6.5. MATERIAŁY DODATKOWE	16
6.6. ĆWICZENIA	16
7. MONITOROWANIE I KONTROLA TESTÓW	18
7.1. TEMATY	18
7.2. CELE	18
7.3. WYNIKI	18
7.4. MATERIAŁY	18
7.5. MATERIAŁY DODATKOWE	18
7.6. ĆWICZENIA	19

Licencja
publiczna



Niniejsza publikacja © 2023 opracowana przez the Software Testing Consortium Partners jest na licencji Attribution-NonCommercial-NoDerivatives 4.0 International. Licencja dostępna pod adresem:
<http://creativecommons.org/licenses/by-nc-nd/4.0/>

0. Obszar „Moje otoczenie”

Witamy na drugim etapie ścieżki edukacyjnej „Testowanie zwinne”.

Dzięki obszarowi „Ja” dowiedziałeś się, iż **wszyscy jesteśmy testerami w naszym codziennym życiu**, nie zdając sobie nawet z tego sprawy. Świadomość ta, miejmy nadzieję, będzie ci towarzyszyć i zachęcać do postrzegania swojego otoczenia przez pryzmat początkującego testera. Rozpoznasz przypadki, w których nieświadomie przeprowadziłeś testy i to wielokrotnie. W rezultacie zaczniesz postrzegać aplikacje na swoim urządzeniu mobilnym z zupełnie nowej perspektywy.

Dzięki **obszarowi „Moje środowisko”** **pogłębisz swoją wiedzę na temat błędów** i lepiej zrozumiesz, w jaki sposób skutecznie je zgłaszać. **Dowiesz się również, jak pisać testy** i jak dostosować je do wymogów aplikacji.



1. Planowanie testów

1.1. Tematy



Podstawy testowania

- Po co testować?
- Kiedy należy testować?
- Zapewnienie jakości

Przegląd rodzajów testów

- Rodzaje i przyczyny usterek.

1.2. Cele

Jako przyszły tester będziesz w stanie zdefiniować cel testu w trakcie procesu jego opracowywania, a także zidentyfikować różne rodzaje testów i usterek, aby podjąć uzasadnione decyzje jako tester.



1.3. Wyniki



Po zakończeniu tego działu będziesz mógł:

- Wyjaśnić, kiedy testy powinny być przeprowadzane w trakcie procesu opracowywania.
- Określić kategorie klasyfikacji usterek.
- Określić potencjalne przyczyny usterek.

1.4. Materiały

- Testowanie oprogramowania i typy testów: definicje <https://www.guru99.com/software-testing-introduction-importance.html> (tekst łącznie z video, 2 min.)



- Różnica między błędem (bug), usterką, wadą, awarią i błędami (errors) <https://www.geeksforgeeks.org/software-testing-bug-vs-defect-vs-error-vs-fault-vs-failure/> (tekst)
- Rodzaje błędów oprogramowania: <https://www.softwaretestinghelp.com/types-of-software-errors/> (tekst)

- Rodzaje wad oprogramowania:

<https://www.testingmind.com/20-types-of-software-defects-every-tester-should-know/> (tekst)

- Rodzaje usterek oprogramowania:

<https://www.educative.io/answers/what-are-faults-in-software-engineering> (tekst)



1.5. Materiały dodatkowe

- Dlaczego testuję:

<https://www.indiumsoftware.com/blog/why-software-testing/> (tekst)

1.6. Ćwiczenia

Po zapoznaniu się z materiałami spróbuj wykonać kilka ćwiczeń, aby wykorzystać nabytą wiedzę w praktyce.



Ćwiczenie 1:

Dlaczego testy uznaje się za ważny krok w procesie opracowywania oprogramowania?

Ćwiczenie 2:

- Proszę wyjaśnić różnicę między błędem (bug), usterką, wadą, awarią i błędami (errors).

.

Ćwiczenie 3:

- Podaj przykłady błędu (bug), usterki, wady, awarii i błędów (errors) oprogramowania.

2. Analiza testów

2.1. Tematy



- Podstawa testów i obiekty testowe
- Historyjki użytkownika

2.2 Cele

Jako przyszły tester będziesz w stanie wykorzystać „Historyjki użytkownika”, aby przedstawić wymagania w zrozumiały i przejrzysty sposób.



Ponadto, będziesz w stanie opisać podstawę testów i obiekty testowe, aby podjąć uzasadnione decyzje jako tester.

2.3. Wyniki



Po zakończeniu tego działu będziesz mógł:

- Sformułować wymagania dla konkretnego procesu.
- Wyjaśnić zastosowanie „Historyjek użytkownika”.
- Napisać „Historyjkę użytkownika” za pomocą Jiry.
- Zdefiniować własnymi słowami termin „obiekt testowy”.

2.4. Materiały

- Obiekty testowe w testowaniu oprogramowania: definicje

<https://en.itpedia.nl/2019/10/07/wat-zijn-testobjecten-bij-softwaretesting/> (tekst)



- Historyjki użytkownika:

<https://www.mountaingoatsoftware.com/agile/user-stories#:~:text=User%20stories%20are%20short%2C%20simple,so%20that%20> (tekst)

- Kryteria dotyczące historyjek użytkownika i akceptacji: Przykłady

<https://www.softwaretestinghelp.com/user-story-acceptance-criteria/> (tekst)



2.5. Materiały dodatkowe

- Jak pisać historyjkę użytkownika i kryteria akceptacji w JIRA: <https://www.youtube.com/watch?v=XPqOqWw0d0c>
(Video)

2.6. Ćwiczenia

Po zapoznaniu się z materiałami spróbuj wykonać kilka ćwiczeń, aby wykorzystać nabytą wiedzę w praktyce.



Ćwiczenie 1:

Zdefiniuj termin „obiekt testowy” własnymi słowami.

Ćwiczenie 2:

Podaj co najmniej 3 przykłady potencjalnych obiektów testowych.

Ćwiczenie 3:

Wyjaśnij korzyści stosowania historyjek użytkownika.

Zadanie 4:

Spróbuj przetestować konkretną stronę internetową. Powinieneś znać tę stronę, może nią być strona twojego ulubionego sklepu internetowego, z którego korzystasz od czasu do czasu. Przed rozpoczęciem procesu testowania, pomyśl o konkretnym procesie, który chcesz przetestować (np. dokonywanie opłaty, dodawanie towarów do koszyka, itp.).

- Wybierz proces i opisz go.
- Następnie opisz szczegółowo, w jaki sposób, jako użytkownik, spodziewasz się, że proces ten będzie działał na każdym etapie – w ten sposób określisz swoje wymagania. Spróbuj je opisać bardzo precyzyjnie, dzieląc na małe etapy.
- Ponownie zapoznaj się z wymogami, które sformułowałeś w odniesieniu do strony internetowej. Być może przypominają już one historyjki

użytkownika lub nadal wyglądają nieco inaczej – spróbuj dostosować je do „Historyjek użytkownika”.

- o Teraz przejdź do wdrażania historyjek użytkownika do JIRY. (W razie problemów związanych z JIRA, skontaktuj się ze swoim trenerem.)

WSKAZÓWKA: Narzędziem powszechnie używanym do obsługi historyjek użytkownika jest JIRA. Jeśli wcześniej nie pracowałeś/-aś z JIRA, tu możesz znaleźć krótkie wprowadzenie.

<https://www.youtube.com/watch?v=XPqOqWw0d0c>

(Wideo 14:00)

Link do pobierania danych JIRA: <https://www.atlassian.com/>

3. Projektowanie testów

3.1. Tematy



- Przypadki testowe

3.2. Cele

Jako przyszły tester będziesz w stanie napisać przypadki testowe, które stanowią podstawę dobrej dokumentacji każdego projektu informatycznego.



3.3. Wyniki



Po zakończeniu tego działu będziesz mógł:

- Napisać przypadki testowe w odniesieniu do każdego projektu.
- Opisać własnymi słowami, czym jest „przypadek testowy”.

3.4. Materiały

Pierwsze trzy linki dotyczą podobnych tematów. Nie musisz zapoznawać się ze wszystkimi, wybierz te, które ci odpowiadają.



- Czym jest projektowanie testów:
<http://tryqa.com/what-is-test-design-when-to-create-it/> (tekst)
- Przypadki testowe: rodzaje, znaczenie, najlepsze praktyki:
<https://searchsoftwarequality.techtarget.com/definition/test-case>
(Tekst, w tym wideo)
- Pisanie przypadków testowych w testowaniu oprogramowania:
<https://www.guru99.com/test-case.html> (Tekst; w tym nagranie video)



3.5. Ćwiczenia

Po zapoznaniu się z materiałami spróbuj wykonać kilka ćwiczeń, aby wykorzystać nabytą wiedzę w praktyce.

Ćwiczenie 1:

Zdefiniuj termin „przypadek testowy” w maksymalnie trzech zdaniach.

Ćwiczenie 2:

Na podstawie przeanalizowanych materiałów spróbuj napisać pierwsze przypadki testowe. Wybierz funkcję strony internetowej, z której lubisz korzystać, i napisz przypadek testowy, korzystając z ulubionych narzędzi. Dostępne narzędzia to np: Word, JIRA, Rail test, itp.

4. Implementacja testów

4.1. Tematy



- Testowanie eksploracyjne

4.2. Cele

Jako przyszły tester będziesz w stanie wykorzystać testowanie eksploracyjne do rozwiązywania rzeczywistych problemów, co zwiększy kreatywność i zachęci Cię do wybierania nieoczywistych rozwiązań.



4.3 Wyniki



Po zakończeniu tego działu będziesz mógł:

- Zdefiniować własnymi słowami testowanie eksploracyjne.
- Zastosować testowanie eksploracyjne do znanego produktu.

4.4. Materiały

- Testowanie eksploracyjne: Definicje, najlepsze praktyki, techniki, wyzwania

<https://www.guru99.com/exploratory-testing.html> (tekst)

- Eksploracyjny charakter testowania oprogramowania:

<https://www.satisfice.com/exploratory-testing> (Tekst; Wykład)

- Wykład: Sesja testowania eksploracyjnego Jamesa Basha

<https://www.satisfice.com/download/session-based-test-management>

(wykład)



4.5. Materiały dodatkowe

- Testowanie eksploracyjne:

<https://www.youtube.com/watch?v=6TcAg3a82Xw> (video,

6 min.)



4.6. Ćwiczenia

Po zapoznaniu się z materiałami spróbuj wykonać kilka ćwiczeń, aby wykorzystać nabytą wiedzę w praktyce.

Ćwiczenie 1:

Zdefiniuj termin „testowanie eksploracyjne” w maksymalnie pięciu zdaniach.

Ćwiczenie 2:

Jakie są twoim zdaniem zalety testowania eksploracyjnego? Wymień co najmniej 3 zalety i krótko je scharakteryzuj.

Ćwiczenie 3:

A teraz weź długopis leżący na biurku, i... zacznij go testować. Wykonaj WSZYSTKIE możliwe czynności związane z testowaniem!

Poddawaj testowi wszystko co możliwe, spisz je, wyjaśniając, co dokładnie zrobiłeś i jak wyglądają Twoje wyniki.

Pamiętaj, iż ćwiczenie to jest ograniczone czasowo i możesz mu poświęcić maksymalnie 40 minut.

5. Wykonanie Testów

5.1. Tematy



- Błędy
- Testowanie a debugowanie

5.2. Cele

Jako przyszły tester będziesz w stanie wyjaśnić termin „błąd” i prawidłowo go zidentyfikować.



5.3. Wyniki



Po zakończeniu tego działu będziesz mógł:

- Zdefiniować własnymi słowami termin „błąd”.
- Podać różnicę między „testowaniem” a „debugowaniem”.

5.4. Materiały

- Błędy: definicja i rodzaje błędów

<https://academy.test.io/en/articles/2541938-types-of-bugs> (tekst i video, 2 min.)



- Klasyfikacja błędów w testowaniu oprogramowania: Do czego jest potrzebna?

<https://www.scnsoft.com/software-testing/types-of-bugs> (tekst)

- Testowanie i debugowanie: różnice

<https://testsigma.com/blog/difference-between-testing-and-debugging/> (tekst)



5.5. Materiały dodatkowe

- Charakterystyka: testowanie a debugowanie:

<https://www.youtube.com/watch?v=D66hHoxMz5M> (video, 2 min.)

- Wykrywanie błędów:

<https://youtu.be/3VY2SL6ee40?si=yJkHfieu-wv9Vple> (video, 6 min.)

5.6. Ćwiczenia

Po zapoznaniu się z materiałami spróbuj wykonać kilka ćwiczeń, aby wykorzystać nabytą wiedzę w praktyce.



Ćwiczenie 1:

Jak zdefiniujesz własnymi słowami termin „błąd”?

Ćwiczenie 2:

Wyjaśnij własnymi słowami różnicę między testowaniem a usuwaniem błędów.

6. Zamknięcie testów

6.1. Tematy



- Dokumentacja testu

6.2. Cele

Jako przyszły tester będziesz w stanie stworzyć przejrzystą dokumentację, aby zapewnić sprawną i skuteczną współpracę w zespole i przekazywanie zadań.



6.3. Wyniki



Po zakończeniu tego działu będziesz mógł:

- Wymienić korzyści wynikające z dobrze prowadzonej dokumentacji.
- Tworzyć przejrzystą i zrozumiałą dokumentację.

6.4. Materiały

- Raport z zamknięcia testu:

<https://testsigma.com/blog/test-closure-report/> (tekst)



- Jak pisać przypadki testowe:

<https://www.softwaretestinghelp.com/how-to-write-effective-test-cases-test-cases-procedures-and-definitions/> (tekst)



6.5. Materiały dodatkowe

- Dokumentacja testu:

https://youtu.be/LO9Hs09BbZQ?si=_sbHY4yONo7q-gj2 (video 6 min.)

6.6. Ćwiczenia

Po zapoznaniu się z materiałami spróbuj wykonać kilka ćwiczeń, aby wykorzystać nabytą wiedzę w praktyce.



Ćwiczenie 1:

Poproś trenera o przekazanie informacji zwrotnych na temat stworzonych przez ciebie przypadków testowych. **(Rozdział 3. Projektowanie testów, ćwiczenie 2).**

Ćwiczenie 2:

Dlaczego twoim zdaniem ważne jest dokumentowanie przypadków testowych i wyników procesów testowania? Jakie są zalety dokumentacji? Napisz o swoich przemyśleniach (maksymalnie połowa strony).

7. Monitorowanie i kontrola testów

7.1. Tematy



- Rodzaje ryzyka związane z produktem i projektem oraz sposób ich identyfikacji.

7.2. Cele

Jako przyszły tester będziesz w stanie zidentyfikować rodzaje ryzyka związanego z produktem i projektem oraz opracować strategię ich zapobiegania.



7.3. Wyniki



Po zakończeniu tego działu będziesz mógł:

- Określić rodzaje ryzyka związane z produktem i projektem oraz opracować strategię ich skutecznego unikania.

7.4. Materiały

- Monitorowanie i kontrola testów w testowaniu oprogramowania:

<https://www.guru99.com/how-you-can-achieve-project-goals-through-test-monitoring-control.html> (tekst)



- Ryzyko związane z produktem i ryzyko związane z projektami:

<https://qakumar.wordpress.com/2018/03/08/product-risk-vs-project-risk/> (tekst)

- Różnice między ryzykiem związanym z produktem a ryzykiem związanym z projektem:

<https://qablog.practitest.com/the-simple-differences-between-product-risks-project-risks/> (tekst)



7.5. Materiały dodatkowe

- Charakterystyka: Rodzaje ryzyka, ryzyko związane z produktem, ryzyko związane z projektem:

<https://www.youtube.com/watch?v=AZ7M877j2Pw> (video, 9 min.)

7.6. Ćwiczenia

Po zapoznaniu się z materiałami spróbuj wykonać kilka ćwiczeń, aby wykorzystać nabytą wiedzę w praktyce.



Ćwiczenie 1:

Wyjaśnij własnymi słowami okoliczności, w których konieczne jest przeprowadzenie testu, i dlaczego jest to tak istotne.

Ćwiczenie 2:

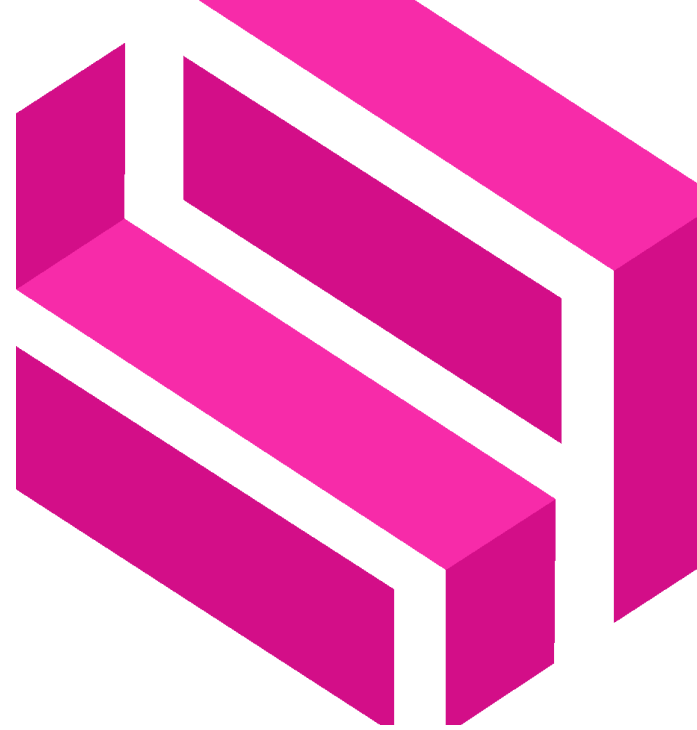
Pomyśl o ryzyku, które mogłyby zagrozić projektowi, i opisz je.

Ćwiczenie 3:

Wyjaśnij, czym jest „ryzyko związane z produktem”, i podaj przykłady.

Zadanie 4:

Wymień niektóre środki stosowane w ramach monitorowania projektów.



SOFTWARE TESTING ACADEMY

Testowanie zwinne (Agile)

Część 3: Obszar „Mój zespół”

R2 – Materiały szkoleniowe

Projekt The Software Testing Academy został zrealizowany przy wsparciu finansowym Unii Europejskiej. Niniejsza publikacja odzwierciedla jedynie stanowisko autora, a Komisja nie ponosi odpowiedzialności za ewentualne wykorzystanie zawartych w niej informacji.

Autorzy: Amaris Consulting – <https://www.amaris.com/>



Co-funded by
the European Union

Spis treści

0. OBSZAR „MÓJ ZESPÓŁ”	4
1. PLANOWANIE TESTÓW	5
1.1. TEMATY	5
1.2. CELE	5
1.3. REZULTAT.....	5
1.4. MATERIAŁY	5
1.5. MATERIAŁY DODATKOWE	6
1.6. ĆWICZENIA	6
2. ANALIZA TESTÓW	7
2.1. TEMATY	7
2.2. CELE	7
2.3. WYNIKI.....	7
2.4. MATERIAŁY	7
2.5. ĆWICZENIA	7
3. PROJEKTOWANIE TESTÓW.....	9
3.1. TEMATY	9
3.2. CELE	9
3.3. WYNIKI.....	9
3.4. MATERIAŁY	9
3.5. ĆWICZENIA	10
4. IMPLEMENTACJA TESTÓW	11
4.1. TEMATY	11
4.2. CELE	11
4.3. WYNIKI.....	11
4.4. MATERIAŁY	11
4.5. MATERIAŁY DODATKOWE	12
4.6. ĆWICZENIA	12
5. WYKONANIE TESTÓW	13
5.1. TEMATY	13
5.2. CELE	13
5.3. WYNIKI.....	13
5.4. MATERIAŁY	13
5.4.1. Zgłaszanie błędów.....	13
5.4.2. Informacja zwrotna	14
5.4.3. Aktywne słuchanie	14

5.4.4.	Praca w zespole.....	14
5.5.	MATERIAŁY DODATKOWE	15
5.6.	ĆWICZENIA	15
5.6.1.	Praca zespołowa	15
5.6.2.	Aktywne słuchanie	15
5.6.3.	Informacja zwrotna	16
5.6.4.	Zgłaszanie błędów.....	16
6.	ZAMKNIĘCIE TESTÓW	17
6.1.	TEMATY	17
6.2.	CELE	17
6.3.	WYNIKI.....	17
6.4.	MATERIAŁY	17
6.5.	ĆWICZENIA	17
7.	MONITOROWANIE I KONTROLA TESTÓW	19
7.1.	TEMATY	19
7.2.	CELE	19
7.3.	WYNIKI.....	19
7.4.	MATERIAŁY	19
7.5.	ĆWICZENIA	19

Licencja publiczna



Niniejsza publikacja © 2023 opracowana przez the Software Testing Consortium Partners jest na licencji Attribution-NonCommercial-NoDerivatives 4.0 International. Licencja dostępna pod adresem: <http://creativecommons.org/licenses/by-nc-nd/4.0/>

0. Obszar „Mój zespół”

Witamy na trzecim etapie ścieżki edukacyjnej „Testowanie zwinne”.

Obszar „Mój zespół” poświęcony jest **wykształceniu umiejętności wspólnego testowania w obrębie zespołu**. Innymi słowy, w tej części zapoznasz się z istotnymi elementami niezbędnymi do skutecznej współpracy w ramach zespołu testującego. Poznasz podstawy stałej współpracy i wzajemnego rozwoju w zakresie opracowywania i oprogramowania.



testowania

1. Planowanie testów

1.1. Tematy



- Cele testów
- Typy testów
- Techniki testowe (Jak z nich korzystać?)

1.2. Cele

Jako przyszły tester będziesz w stanie dokonać podsumowania najważniejszych celów testowania oraz prawidłowo wybrać i zastosować techniki testowania do rozwiązywania problemów w celu przeprowadzenia bardziej ukierunkowanych i uzasadnionych testów.



1.3. Rezultat



Po zakończeniu tego działu będziesz mógł:

- Opowiedzieć własnymi słowami o poszczególnych rodzajach testów.
- Prawidłowo przypisać zadania sytuacyjne do odpowiednich typów testów.
- Dobrać odpowiednie techniki testowania do konkretnych sytuacji/problemów.

1.4. Materiały

- Cele testowania oprogramowania:

<https://www.toolsqa.com/software-testing/istqb/software-testing-objectives/> (tekst)



- Cel i założenia testowania oprogramowania:

<https://www.qamadness.com/what-is-the-ultimate-goal-and-key-objectives-of-software-testing/> (tekst)

- Rodzaje testowania oprogramowania:

<https://www.softwaretestinghelp.com/types-of-software-testing/> (tekst)



1.5. Materiały dodatkowe

- Rodzaje testowania oprogramowania:
<https://www.guru99.com/types-of-software-testing.html>
(tekst)

- 5 rodzajów testowania oprogramowania:
<https://youtu.be/YaXJeUkBe4Y?si=jKnszYpolGK6ZNIa> (video, 6:24 min.)

1.6. Ćwiczenia

Po zapoznaniu się z materiałami spróbuj wykonać kilka ćwiczeń, aby wykorzystać nabytą wiedzę w praktyce.



Ćwiczenie 1:

Wymień 5 rodzajów testów i opisz każdy z nich własnymi słowami (w jednym lub dwóch zdaniach dla każdego z nich).

Ćwiczenie 2:

Wyobraź sobie, iż twoim zadaniem jest przetestowanie narzędzia komunikacji na rynku telefonii komórkowej. Wszystkie rozmowy telefoniczne nie są przełączane przez pojedyncze urządzenia, lecz przechodzą przez serwer centralny. Zadaniem użytkownika jest przetestowanie oprogramowania serwerowego. Wymień rodzaje testów, które wykorzystasz w procesie testowania, i krótko uzasadnij swój wybór własnymi słowami. *W razie potrzeby przedyskutuj to ćwiczenie z trenerem.*

2. Analiza testów

2.1. Tematy



- Specyfikacja wymogów systemowych
- Specyfikacja wymogów biznesowych

2.2. Cele

Jako przyszły tester będziesz w stanie prawidłowo przeanalizować testy, aby być w stanie dostarczyć pożądane wyniki swoim kolegom i klientom.



2.3. Wyniki



Po zakończeniu tego działu będziesz mógł:

- Odpowiednio dobrać techniki testowania do wymagań.

2.4. Materiały

- Specyfikacja wymogów dotyczących oprogramowania:
<https://www.softwaretestinghelp.com/how-to-test-software-requirements-specification-srs/> (tekst)



- Różnice między specyfikacją wymogów biznesowych a specyfikacją wymogów systemowych:
<https://www.geeksforgeeks.org/software-testing-brs-vs-srs/> (tekst)

2.5. Ćwiczenia

Po zapoznaniu się z materiałami spróbuj wykonać kilka ćwiczeń, aby wykorzystać nabytą wiedzę w praktyce.



Ćwiczenie 1:

Czym jest wymóg dotyczący oprogramowania? Wyjaśnij w kilku słowach. Wymień niektóre aspekty będące częścią wymogu dotyczącego oprogramowania.

Ćwiczenie 2:

Czym jest specyfikacja wymogów biznesowych i jakie są jej cele? Wyjaśnij w kilku słowach.

3. Projektowanie testów

3.1. Tematy



- Techniki testowania służące do opracowywania przypadków testowych, w tym testowanie oparte na doświadczeniu i dane testowe.

3.2. Cele

Jako przyszły tester będziesz w stanie wykonać projekt testowy oparty na uzasadnionych, wybranych przypadkach testowych, aby przeprowadzić test w miarę dokładnie ale przy pewnych ograniczeniach.



3.3. Wyniki



Po zakończeniu tego działu będziesz mógł:

- Wyjaśnić proces opracowywania projektowania testów.
- Podać liczbę przypadków testowych potrzebnych do przetestowania produktu i uzasadnić powody takiego wyboru.
- Wyjaśnić termin „dane testowe”.

3.4. Materiały

- Techniki projektowania testów:

<https://www.lotus-qa.com/blog/test-case-design-techniques/> (tekst)

- Dane w testowaniu oprogramowania:

<https://www.guru99.com/software-testing-test-data.html> (tekst)

- Dane testowe, techniki projektowania testów z przykładami:

<https://www.softwaretestinghelp.com/tips-to-design-test-data-before-executing-your-test-cases/> (tekst)



- Testowanie oparte na doświadczeniu:

<https://tryqa.com/what-is-experience-based-testing-technique/> (tekst)

3.5. Ćwiczenia

Po zapoznaniu się z materiałami spróbuj wykonać kilka ćwiczeń, aby wykorzystać nabytą wiedzę w praktyce.



Ćwiczenie 1:

Wyjaśnij własnymi słowami, jakie są techniki testowania, oraz wymień niektóre techniki projektowania testów i ich zalety. Postaraj się być zwięzły się w swojej odpowiedzi.

Ćwiczenie 2:

Podaj przykłady danych testowych wraz z ich wykorzystaniem..

Ćwiczenie 3:

Wyjaśnij, czym są twoim zdaniem testy oparte na doświadczeniu. Postaraj się być zwięzły się w swojej odpowiedzi.

4. Implementacja testów

4.1. Tematy



- Pakiety testów.

4.2. Cele

Jako przyszły tester będziesz w stanie wyjaśnić, w jaki sposób stosuje się pakiety testów, aby w razie potrzeby z nich skorzystać.



4.3. Wyniki



Po zakończeniu tego działu będziesz mógł:

- Zdefiniować własnymi słowami termin „Pakiet testów”.
- Wyjaśnić, kiedy i dlaczego należy skorzystać z pakietu testów.
- Wyjaśnić, w jaki sposób korzystać z pakietu testów.

4.4. Materiały

- Czym jest pakiet testów?



https://www.tutorialspoint.com/software_testing_dictionary/test_suite.htm

(tekst)

- Wykorzystanie pakietów testów i przypadków testowych:

<https://help.testlodge.com/hc/en-us/articles/115007101447-Using-Test-Suites-and-Test-Cases>

(Tekst + przykłady + wideo)

- Instruktaż na temat pakietów testów:

<https://www.lambda-test.com/learning-hub/test-suite>

(tekst)



4.5. Materiały dodatkowe

- Przypadki testowe i pakiety testów:
<https://goo.su/QMHHRV> (tekst)

- Różnice między przypadkami testowymi a pakietami testów:
<https://www.testmonitor.com/blog/test-case-test-suite-test-run-whats-the-difference> (tekst)

4.6. Ćwiczenia

Po zapoznaniu się z materiałami spróbuj wykonać kilka ćwiczeń, aby wykorzystać nabytą wiedzę w praktyce.



Ćwiczenie 1:

Wyjaśnij własnymi słowami czym są pakiety testów (w maksymalnie 3 zdaniach).

Ćwiczenie 2:

Jakie są zalety stosowania pakietów testów? Wymień (w punktach) trzy zalety.

Ćwiczenie 3:

Wyobraź sobie przekazanie pakietu testów kolegom z zespołu. Jakie informacje musiałbyś im przekazać, aby mieć pewność, że będą mogli skutecznie korzystać z pakietu testów? Podaj 5 kluczowych informacji istotnych dla opracowania pakietów testów.

5. Wykonanie testów

5.1. Tematy



- Informowanie o błędach
- Informacja zwrotna (zalety informacji zwrotnej przekazywanej regularnie)
- Praca w zespole (środki usprawniające i aktywne słuchanie)
- Aktywne słuchanie zapewniające rzetelne zgłaszanie błędów.

5.2. Cele

Jako przyszły tester będziesz w stanie uważniej i skuteczniej komunikować się z innymi, aby usprawnić pracę w zespole i wyniki pracy.



5.3. Wyniki



Po zakończeniu tego działu będziesz mógł:

- Napisać raport na temat błędu, ogólnie zrozumiały dla wszystkich.
- Określić niektóre kluczowe aspekty udanej pracy zespołowej.
- Wyjaśnić, co oznacza „aktywne słuchanie” i streścić je własnymi słowami.
- Zastanowić się nad pewnymi inicjatywami które można wdrożyć, aby usprawnić pracę w zespole.
- Zrozumieć jak ważne jest regularne przekazywanie swojemu zespołowi informacji zwrotnej.

5.4. Materiały

5.4.1. Zgłaszanie błędów

- Na czym polega zgłaszanie błędów?

[-https://www.guru99.com/defect-management-process.html](https://www.guru99.com/defect-management-process.html) (tekst)



- Jak dokonać zgłoszenia błędów: dwa sposoby.
<https://www.softwaretestinghelp.com/how-to-write-good-bug-report/> (tekst)
<https://www.softwaretestinghelp.com/sample-bug-report/> (tekst)

5.4.2. Informacja zwrotna

- Metodyka przekazywania informacji zwrotnej:
<https://positivepsychology.com/negative-feedback/> (Tekst; Filmy video)
- Pięć sposobów przekazywania negatywnej informacji zwrotnej:
<https://plan.io/blog/negative-feedback/> (tekst)
- Przekazywanie konstruktywnej informacji zwrotnej:
<https://corporatefinanceinstitute.com/resources/careers/soft-skills/constructive-feedback/> (tekst)
 - Zalety informacji zwrotnej przekazywanej regularnie
<https://engagedly.com/continuous-feedback-benefits/> (tekst)
- Pozytywna informacja zwrotna: znaczenie sposobu przekazania
<https://www.indeed.com/career-advice/career-development/importance-of-positive-feedback> (tekst)

5.4.3. Aktywne słuchanie

- Aktywne słuchanie: sztuka empatycznej konwersacji
<https://positivepsychology.com/active-listening/> (tekst)
- Aktywne słuchanie: Definicja, umiejętności i przykłady:
<https://www.thebalancecareers.com/active-listening-skills-with-examples-2059684> (tekst)

5.4.4. Praca w zespole

- Praca w zespole: Siedem cech
<https://theundercoverrecruiter.com/qualities-successful-work-team/> (tekst)

- Praca w zespole:

<https://rework.withgoogle.com/print/guides/5721312655835136/> (tekst)

- Środki usprawniające pracę zespołu w miejscu pracy:

<https://blog.jostle.me/blog/12-easy-ways-to-improve-workplace-teamwork>
(tekst)



5.5. Materiały dodatkowe

- Udana praca zespołowa w Google:

<https://www.youtube.com/watch?v=v2PaZ8NI2T4>

(video,

4 min.)

5.6. Ćwiczenia

Po zapoznaniu się z materiałami spróbuj wykonać kilka ćwiczeń, aby wykorzystać nabytą wiedzę w praktyce.



5.6.1. Praca zespołowa

Ćwiczenie 1:

Czy zgadzasz się z czynnikami niezbędnymi do osiągnięcia sukcesu wymienionymi w materiałach dotyczących pracy w zespole? Jeżeli tak, to dlaczego? Jeżeli nie, to dlaczego? Wybierz cztery aspekty czynników niezbędnych do osiągnięcia sukcesu i krótko je scharakteryzuj.

Ćwiczenie 2:

Zastanów się, w jaki sposób możesz przyczynić się do zwiększenia motywacji w zespole i poprawy pracy w zespole w swoim otoczeniu.

5.6.2. Aktywne słuchanie

Ćwiczenie 1:

Jakie są twoim zdaniem cechy aktywnego słuchania i w jaki sposób mogą być one korzystne dla pracy w zespole?

Ćwiczenie 2:

Spróbuj wyjaśnić komuś ze swojego otoczenia tę ścieżkę nauki. Czy ta osoba aktywnie słuchała? Jak to ustaliłeś?

5.6.3. Informacja zwrotna

Ćwiczenie 1:

Wymień 3 kluczowe korzyści płynące z regularnego przekazywania informacji zwrotnej.

Ćwiczenie 2:

Jakie jest twoje zdanie na temat tej ścieżki szkolenia? W jaki sposób ją udoskonalilibyś? Jakie są jej zalety?

5.6.4. Zgłaszanie błędów

Ćwiczenie 1:

Poćwiczmy teraz pisanie raportu na temat błędów! Wybierz stronę internetową, którą lubisz np. kanał YouTube, e-commerce, itp., spróbuj znaleźć wszystkie błędy i sporządzić raport.

6. Zamknięcie testów

6.1. Tematy



- Dokumentacja procesu testowania

6.2. Cele

Jako przyszły tester będziesz w stanie wyjaśnić jak ważna jest dokumentacja testu w procesie testowania oprogramowania.



6.3. Wyniki



Po zakończeniu tego działu będziesz mógł:

- Wymienić ważne aspekty dobrze prowadzonej dokumentacji w ramach procesów testowych.

6.4. Materiały

- Przewodnik na temat dokumentacji testów:
<https://www.softwaretestinghelp.com/why-documentation-is-important-in-software-testing/> (tekst)



- Znaczenie dokumentacji testowania oprogramowania:
<https://www.testquality.com/blog/tpost/ghkxpe44a1-why-is-test-documentation-necessary-in-s> (tekst)

- Dokumentacja oprogramowania: Najlepsze praktyki
<https://helpjuice.com/blog/software-documentation> (tekst)

6.5. Ćwiczenia

Po zapoznaniu się z materiałami spróbuj wykonać kilka ćwiczeń, aby wykorzystać nabytą wiedzę w praktyce.



Ćwiczenie 1:

Z tej lekcji wyniosłeś wiedzę na temat dokumentacji testowej. Czy uważasz, w oparciu o zdobytą wiedzę, iż prawidłowo zarządzałeś procesem dokumentacji podczas wykonywania ćwiczeń w ramach tego programu szkoleniowego? Możesz teraz dostosować i wprowadzić ulepszenia zgodnie z opanowaną treścią.

7. Monitorowanie i kontrola testów

7.1. Tematy



- Kryteria wyjścia
- Żądanie zmiany

7.2. Cele

Jako przyszły tester będziesz w stanie wyjaśnić, kiedy zakończył się proces testowania i w jaki sposób sformułować żądania zmian w celu dokładniejszego przeprowadzenia testu.



7.3. Wyniki



Po zakończeniu tego działu będziesz mógł:

- Zdefiniować termin „kryteria wyjścia”.
- Sformułować żądanie zmian.

7.4. Materiały

- Kryteria wejścia i wyjścia w testowaniu oprogramowania:
<https://www.rishabhsoft.com/blog/entry-and-exit-criteria-in-software-testing> (tekst)



- Kryteria wyjścia: Przykłady
<https://goo.su/xcW9AT> (tekst)

7.5. Ćwiczenia

Po zapoznaniu się z materiałami spróbuj wykonać kilka ćwiczeń, aby wykorzystać nabytą wiedzę w praktyce.

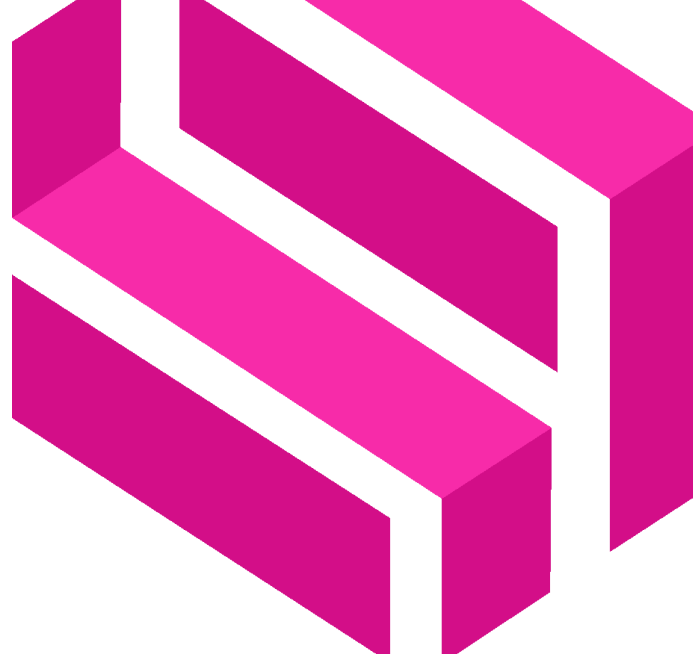


Ćwiczenie 1:

Zdefiniuj własnymi słowami pojęcie „kryteria wyjścia” (w 3–5 zdaniach).

Ćwiczenie 2:

Połączmy teraz temat kryteriów wyjścia z tą Akademią: Jakie kryteria musi spełniać ta Akademia, aby odnieść sukces w nauczaniu? Sporządź listę.



SOFTWARE TESTING ACADEMY

Testowanie zwinne (Agile)

Część 3: Obszar „Moje przedsiębiorstwo”

R2 – Materiały szkoleniowe

Projekt The Software Testing Academy został zrealizowany przy wsparciu finansowym Unii Europejskiej. Niniejsza publikacja odzwierciedla jedynie stanowisko autora, a Komisja nie ponosi odpowiedzialności za ewentualne wykorzystanie zawartych w niej informacji.

Autorzy: Amaris Consulting – <https://www.amaris.com/>



Co-funded by
the European Union

Spis treści

0. OBSZAR „MOJE PRZEDSIĘBIORSTWO”	5
1. PLANOWANIE TESTÓW	6
1.1. TEMATY	6
1.2. CELE	6
1.3. WYNIKI	6
1.4. MATERIAŁY	6
1.4.1 <i>Metodologia Agile i Waterfall</i>	6
1.4.2 <i>Środowiska testowe</i>	7
1.5. MATERIAŁY DODATKOWE	7
1.6. ĆWICZENIA	7
2. ANALIZA TESTÓW	9
2.1. TEMATY	9
2.2. CELE	9
2.3 WYNIK	9
2.4 MATERIAŁY	9
2.5 DODATKOWE MATERIAŁY	9
2.6 ĆWICZENIA	10
3. PROJEKTOWANIE TESTÓW	11
3.1 TEMATY	11
3.2 CELE	11
3.3 WYNIKI	11
3.4 MATERIAŁY	11
3.5 ĆWICZENIA	12
4. IMPLEMENTACJA TESTÓW	13
4.1 TEMATY	13
CELE 4.2.....	13
4.3 WYNIKI	13
4.4 MATERIAŁY	13
4.5 ĆWICZENIA	13
5. WYKONANIE TESTÓW	15
5.1 TEMATY	15
5.2 CELE	15
5.3 WYNIKI	15
5.4 MATERIAŁY	15
5.5 ĆWICZENIA	15
6. ZAMKNIĘCIE TESTÓW	17

6.1 TEMATY	17
6.2 CELE	17
6.3 WYNIKI	17
6.4 MATERIAŁY	17
6.5 MATERIAŁY DODATKOWE	18
6.6 ĆWICZENIA	18
7. MONITOROWANIE I KONTROLA TESTÓW	19
7.1 TEMATY	19
7.2 CELE	19
7.3 WYNIKI	19
.....	19
7.4 MATERIAŁY	19
7.5 ĆWICZENIA	19

Licencja publiczna



Niniejsza publikacja © 2023 opracowana przez the Software Testing Consortium Partners jest na licencji Attribution-NonCommercial-NoDerivatives 4.0 International. Licencja dostępna pod adresem:

<http://creativecommons.org/licenses/by-nc-nd/4.0/>

0. Obszar „Moje przedsiębiorstwo”

Witamy w czwartym obszarze ścieżki edukacyjnej „Testowanie zwinne”!

W tym obszarze „**Moje przedsiębiorstwo**” omówimy zastosowanie testowania oprogramowania w operacjach na poziomie przedsiębiorstwa.



Dowiesz się o strategiach i metodach skutecznego wdrażania praktyk testowania oprogramowania, dostosowanego do złożonego i dynamicznego charakteru wielkoskalowych środowisk biznesowych.

Rozumiejąc, **w jaki sposób można bezproblemowo zintegrować testy na poziomie przedsiębiorstwa**, dążymy do poprawy jakości, niezawodności i osiągnięcia całościowego sukcesu w biznesie.

1. Planowanie testów

1.1. Tematy



- Modele cyklu życia oprogramowania oraz porównanie modelu kaskadowego (Waterfall) z testowaniem zwinnym (Agile)
- Planowanie warunków testowych: Możliwe opcje i cechy
- Infrastruktura (spotkania, flipchart i narzędzia)

1.2 Cele

Jako przyszły tester będziesz w stanie wyjaśnić różne etapy cyklu życia oprogramowania i metody opracowywania oprogramowania, wynikające z nich korzyści i wady w celu dokonania uzasadnionego wyboru modelu, na którym należy się oprzeć.



1.3. Wyniki



Po zakończeniu tego działu będziesz mógł:

- Wyjaśnić, czym jest cykl życia oprogramowania.
- Nazwać sekwencyjne fazy tworzenia/testowania oprogramowania.
- Wyjaśnić własnymi słowami model kaskadowy (Waterfall).
- Wymienić korzyści i wady modelu kaskadowego (Waterfall).
- Wyjaśnić tworzenie oprogramowania zwinnego (Agile).
- Nazwać korzyści i wady podejścia zwinnego (Agile)

1.4. Materiały

1.4.1 Metodologia Agile i Waterfall

Uwaga: Wybraliśmy bardzo podobne czcionki. Nie ma potrzeby szczegółowego analizowania wszystkich tekstów. Przejrzyj materiały i wybierz 3 artykuły związane z metodyką testowania zwinnego i modelu kaskadowego.



- Cykl życia oprogramowania: Pojęcia, definicje:
<https://www.intellectsoft.net/blog/essentials-software-development-life-cycle/> (tekst)
- Sekwencyjne modele testowania oprogramowania (Waterfall and Agile)
<https://www.guru99.com/software-development-life-cycle-tutorial.html>
(Tekst)
- Waterfall a Agile: Właściwe podejście do projektu:
<https://www.seguetech.com/waterfall-vs-agile-methodology/> (tekst)
- Różnice: Testowanie w Agile a testowanie w Waterfall:
<https://www.geeksforgeeks.org/difference-between-agile-testing-and-waterfall-testing/> (tekst)
- „Kiedy korzystać z czego?” Wybierz właściwe podejście do testowania:
<https://www.agile-minds.com/when-to-use-waterfall-when-agile/> (tekst)

1.4.2 Środowiska testowe

- Środowiska testowe: Definicja, rodzaje i najlepsze praktyki:
<https://launchdarkly.com/blog/test-environments-101-definition-types-and-best/> (tekst)
- Środowisko testowe w testowaniu oprogramowania:
<https://www.guru99.com/test-environment-software-testing.html> (tekst)



1.5. Materiały dodatkowe

- Testowanie w Agile a testowanie w Waterfall:
<https://www.youtube.com/watch?v=GzzkpAOxHXs> (video, 9 min.)

1.6. Ćwiczenia

Po zapoznaniu się z materiałami spróbuj wykonać kilka ćwiczeń, aby wykorzystać nabytą wiedzę w praktyce.



Ćwiczenie 1:

Wyjaśnij w jednym lub trzech zdaniach, na czym polega cykl tworzenia oprogramowania.

Ćwiczenie 2:

Wyjaśnij w dwóch lub trzech zdaniach (dla każdego tematu), czym jest model Waterfall i czym jest podejście Agile.

Ćwiczenie 3:

Podaj dwie zalety i dwie wady modelu Waterfall i podejścia Agile.

Zadanie 4:

Wyjaśnij krótko własnymi słowami (w około dwóch zdaniach), w odniesieniu do których zadań zalecałbyś podejście Agile.

Ćwiczenie 5:

Wyobraź sobie, że chcesz skonstruować rower... Wykorzystaj metodologię Agile i Waterfall w procesie konstrukcji i krótko opisz proces w odniesieniu do każdej z nich.

Ćwiczenie 6:

Wymień 5 różnych rodzajów środowisk testowych i ich cele.

2. Analiza testów

2.1. Tematy



- Warunki testowe

2.2. Cele

Jako przyszły tester będziesz mieć wiedzę na temat warunków testowych w celu zapewnienia odpowiednich warunków podczas testowania problemów, które mogą wystąpić podczas realizacji projektów.



2.3 Wynik



Po zakończeniu tego działu będziesz mógł:

- Zdefiniować własnymi słowami termin „warunki testowe”.
- Określić warunki testowe w środowisku pracy.

2.4 Materiały

- Czym jest analiza testu? Podstawa testu? Identyfikacja warunków testowych:

<http://tryqa.com/what-is-test-analysis-or-how-to-identify-the-test-conditions/> (tekst)

- Warunki testowe: Zalety i wady:

<https://cania-consulting.com/2019/11/08/a-test-manager-guide-test-conditions/> (Tekst + video)



2.5 Dodatkowe materiały

- Warunki testowe:

<https://www.youtube.com/watch?v=PPgHhAsk-EA> (video, 6 min.)

2.6 Ćwiczenia

Po zapoznaniu się z materiałami spróbuj wykonać kilka ćwiczeń, aby wykorzystać nabytą wiedzę w praktyce.



Ćwiczenie 1:

Zdefiniuj własnymi słowami termin „warunki testowe” i zapisz odpowiedź w maksymalnie dwóch zdaniach.

Ćwiczenie 2:

W dwóch lub trzech zdaniach wyjaśnij różnicę między warunkami testowymi a scenariuszem testowym.

Ćwiczenie 3:

Wybierz stronę internetową lub aplikację, która posiada funkcję logowania użytkownika. Podaj WSZYSTKIE warunki testowe dla funkcji logowania użytkownika.

3. Projektowanie testów

3.1 Tematy



- Narzędzia i infrastruktura do zarządzania testami

3.2 Cele

Jako przyszły tester będziesz w stanie wymienić narzędzia, z których korzysta twoje przedsiębiorstwo, i pokrótce ocenić infrastrukturę projektu.



3.3 Wyniki



Po zakończeniu tego działu będziesz mógł:

- Zrozumieć znaczenie narzędzi w zarządzaniu testami, automatyzacji, śledzeniu błędów i inżynierii wymagań.

3.4 Materiały

- Znaczenie zarządzaniem testami:

<https://www.functionize.com/blog/importance-of-test-management-in-automation-strategy> (tekst)



- Znaczenie korzystania z narzędzi zarządzania testami:

https://qacraft.com/importance-of-Test-narzędzia_zarządzania/ (Tekst)

- Kryteria wyboru narzędzi zarządzania testami:

<https://testomat.io/blog/10-key-criteria-choose-the-best-test-management-tool-for-your-team/> (tekst)

- GIT:

<https://www.atlassian.com/git/tutorials/what-is-git> (tekst)

- Zautomatyzowana próba z GIT:

<https://coderefinery.github.io/testing/continuous-integration/> (tekst)

3.5 Ćwiczenia

Po zapoznaniu się z materiałami spróbuj wykonać kilka ćwiczeń, aby wykorzystać nabytą wiedzę w praktyce.



Ćwiczenie 1:

Po zapoznaniu się z materiałami na temat narzędzi zarządzania testami posiadasz wiedzę, jak istotne są narzędzia w środowisku testowania oprogramowania. Wymień co najmniej jedno narzędzie, które można wykorzystać do zarządzania następującym zadaniem:

- Określenie wymagań
- Planowanie testów
- Historyjki użytkowników
- Automatyzacja testów
- Zgłaszanie błędów
- Śledzenie usterek
- CI/CD

W razie potrzeby możesz skontaktować się ze swoim trenerem i omówić z nim tematy.

4. Implementacja testów

4.1 Tematy



- Przykłady automatyzacji testów

Cele 4.2

Jako przyszły tester będziesz w stanie odróżnić automatyzację testów od testowania ręcznego, aby dokonać właściwego wyboru procedur w przyszłych projektach.



4.3 Wyniki



Po zakończeniu tego działu będziesz mógł:

- Podjąć decyzję, które testy należy zautomatyzować.

4.4 Materiały

- Budowanie strategii zautomatyzowanego testowania: Przykłady

<https://performancelabus.com/automated-testing-strategy-how-to-build-examples/> (tekst)



- Przykład automatycznej serii procesów ciągłego testowania (pipeline'u):

<https://dev-tester.com/a-real-world-example-of-an-automated-test-pipeline/> (tekst)

4.5 Ćwiczenia

Po zapoznaniu się z materiałami spróbuj wykonać kilka ćwiczeń, aby wykorzystać nabytą wiedzę w praktyce.



Ćwiczenie 1:

Po zapoznaniu się z zagadnieniem automatyzacji testów posiadasz wiedzę na temat znaczenia automatyzacji testów i warunków jego zastosowania. Jakie

kryteria wykorzystałbyś do rozwiązania problemu dotyczącego automatyzacji testów? Przygotuj co najmniej 4 pytania, które zadasz sobie przed podjęciem decyzji, czy automatyzacja powinna mieć zastosowanie czy też nie.

5. Wykonanie testów

5.1 Tematy



- Specyficzne techniki testowe
- Protokoły testowania zapewniające bezpieczeństwo testów

5.2 Cele

Jako przyszły tester będziesz w stanie wyjaśnić, w jaki sposób stosować różne techniki testowe i jak ważne są protokoły testowania w celu przeprowadzenia ukierunkowanych i miarodajnych testów.



5.3 Wyniki



Po zakończeniu tego działu będziesz mógł:

- Stosować specyficzne techniki testowe w celu rozwiązywania problemów.
- Wyjaśnić znaczenie protokołów testowania

5.4 Materiały

- Techniki projektowania przypadków testowych – wariant 1: <https://testsigma.com/blog/test-case-design-techniques/> (tekst)
- Techniki projektowania przypadków testowych – wariant 2: <https://goo.su/zzlfQ> (tekst)
- Wybór najlepszej techniki testowej: <https://www.softwaretestingclass.com/how-to-choose-which-testing-technique-is-best/> (tekst)



5.5 Ćwiczenia

Po zapoznaniu się z materiałami spróbuj wykonać kilka ćwiczeń, aby wykorzystać nabytą wiedzę w praktyce.



Ćwiczenia te koncentrują się na opanowaniu wiedzy, jak korzystać z różnych technik testowych w celu opracowania odpowiednich przypadków testowych.

Ćwiczenie 1:

Wymień 3 techniki projektowania testów.

Ćwiczenie 2:

W poprzednim ćwiczeniu wymieniałeś techniki projektowania testów. W jakich sytuacjach należy je stosować?

Ćwiczenie 3:

Wybierz stronę internetową hotelu, na której można dokonać rezerwacji w celu przetestowania projektowania przypadków testowych. Skoncentrujemy się na zaprojektowaniu przypadku związanego z **rezerwacją pokoju przez użytkownika**. **Wymień wszystkie kroki, jakie musi podjąć użytkownik: zarejestrowanie się na stronie internetowej w celu dokonania płatności za rezerwację i otrzymanie wiadomości e-mail z potwierdzeniem.** Pamiętaj o następujących aspektach:

- Wymagania dotyczące przeprowadzenia testu:
 - Dane testowe w celu zalogowania się na stronie internetowej, takie jak: e-mail, numer telefonu, itp.
 - Klient musi być zalogowany na stronie internetowej.
 - Płatność musi być dokonywana kartą.
- Oczekiwany wynik: Proces rezerwacji powinien zostać zakończony, a użytkownik powinien otrzymać potwierdzenie pocztą elektroniczną.

WSKAZÓWKA: Zadanie to można wykonać za pomocą Jira, TestRail lub za pomocą narzędzia tekstowego, takiego jak Word, WordPad, notepad, itp. Wybierz najbardziej dogodny dla siebie narzędzie.

Zadanie 4:

Jakie techniki projektowania testów zastosowałbyś w poprzednim zadaniu?

6. Zamknięcie testów

6.1 Tematy



- Doskonalenie testów
- Metody dokumentowania archiwizacji i przekazania projektu

6.2 Cele

Jako przyszły tester będziesz w stanie wymienić techniki, w tym metody dokumentowania służące usprawnieniu testów.



6.3 Wyniki



Po zakończeniu tego działu będziesz mógł:

- Wymienić środki służące usprawnieniu procesów testowania.
- Wymienić metody dokumentowania.

6.4 Materiały

- Usprawnianie procesu testowania:

<https://www.guru99.com/a-complete-guide-to-test-process-improvement.html> (tekst)



- Techniki usprawniania:

- Testowanie przez tłum:

<https://automationhacks.io/2020/10/27/mob-testing-bootstrap-whole-team-quality/> (tekst)

- Karta testów:

<https://www.qualitestgroup.com/insights/technical-hub/how-to-write-an-exploratory-test-charter/> (tekst)

- Testowanie eksploracyjne:

<https://www.tmap.net/wiki/exploratory-testing-et> (tekst)

- Udoskonalenie strategii: Testowanie oparte na mapowaniu myśli:

<https://www.infoq.com/articles/mindmap-driven-testing/> (tekst)



6.5 Materiały dodatkowe

- Społeczności praktyków:
<https://www.scaledagileframework.com/communities-of-practice/> (tekst)

- Testowanie poprzez tłum:
<https://www.testlearning.net/en/posts/mob-testing> (tekst)

- Proces usprawnienia:
<https://betterqa.co/test-improvement-processes/> (tekst)

6.6 Ćwiczenia

Po zapoznaniu się z materiałami spróbuj wykonać kilka ćwiczeń, aby wykorzystać nabytą wiedzę w praktyce.



Ćwiczenie 1:

Którą z metod udoskonalenia testów uważasz za szczególnie interesującą i dlaczego? Uzasadnij swój wybór własnymi słowami w około 5 zdaniach.

Ćwiczenie 2:

Następnie podczas kolejnej sesji ze swoim trenerem przedyskutuj tę, która jest dla Ciebie istotna.

Ćwiczenie 3:

Wymień co najmniej trzy metody testowania eksploracyjnego.

Zadanie 4:

Utwórz mapę myśli w celu udokumentowania i usystematyzowania wszystkich tematów/treści, które opanowałeś do tej pory.

7. Monitorowanie i kontrola testów

7.1 Tematy



- Wskaźniki monitorowania testów
- Rodzaje ryzyka związane z testowaniem

7.2 Cele

Jako przyszły tester będziesz w stanie wymienić najważniejsze rodzaje ryzyka związanego z testowaniem i zastosować odpowiednie wskaźniki monitorowania testów w celu zapewnienia skuteczności procesu testowania.



7.3 Wyniki



Po zakończeniu tego działu będziesz mógł:

- Dobrać odpowiedni wskaźnik monitorowania testów do konkretnego przykładu.
- Wymienić najważniejsze rodzaje ryzyka związane z testowaniem i przedstawić propozycje dotyczące sposobów ich unikania.

7.4 Materiały

- Zarządzanie ryzykiem: Rodzaje i ocena
<https://www.softwaretestinghelp.com/types-of-risks-in-software-projects/> (tekst)



- Identyfikacja i zarządzanie ryzykiem związanym z testowaniem oprogramowania:
<https://smartbear.com/blog/software-testing-risks/> (tekst)

7.5 Ćwiczenia

Po zapoznaniu się z materiałami spróbuj wykonać kilka ćwiczeń, aby wykorzystać nabytą wiedzę w praktyce.



Ćwiczenie 1:

Sporządź listę pięciu rodzajów ryzyka związanego z testowaniem. W jaki sposób twoim zdaniem można ich uniknąć?

Ćwiczenie 2:

W rozdziale 5. Wykonanie, w ćwiczeniu 3 tego modułu wybrałeś funkcję strony internetowej rezerwacji hotelu do opracowania przypadku testowego. Rozważ potencjalne zagrożenia związane z tym procesem oraz zaproponuj, w jaki sposób można ich uniknąć lub je zminimalizować (co najmniej trzy rodzaje ryzyka).

W razie potrzeby możesz przedyskutować je ze swoim trenerem.



SOFTWARE TESTING ACADEMY

Testowanie zwinne (Agile)

Część 5: Obszar „Mój pierwszy projekt”

R2 – Materiały szkoleniowe

Projekt The Software Testing Academy został zrealizowany przy wsparciu finansowym Unii Europejskiej. Niniejsza publikacja odzwierciedla jedynie stanowisko autora, a Komisja nie ponosi odpowiedzialności za ewentualne wykorzystanie zawartych w niej informacji.

Autorzy: Amaris Consulting – <https://www.amaris.com/>



Co-funded by
the European Union

Spis treści

0. OBSZAR „MÓJ PIERWSZY PROJEKT”	5
1. PLANOWANIE TESTÓW	6
1.1 TEMATY	6
1.2 CELE	6
1.3 WYNIKI.....	6
1.4 MATERIAŁY	6
1.5 ĆWICZENIA	7
2. ANALIZA TESTÓW	8
2.1 TEMATY	8
2.2 CELE	8
2.3 WYNIKI.....	8
2.4 MATERIAŁY	8
2.4.1 Analiza produktu.....	8
2.4.2 Scenariusze i warunki testowe.....	9
2.5 DODATKOWE MATERIAŁY.....	9
2.6 ĆWICZENIA	9
3. PROJEKTOWANIE TESTÓW.....	11
3.1 ZAGADNIENIA	11
3.2 CELE	11
3.3 WYNIKI.....	11
3.4 MATERIAŁY	11
3.4.1 Architektura testów	11
3.4.2 Narzędzia do zarządzania i przeprowadzania testów	12
3.5 DODATKOWE MATERIAŁY.....	12
3.6 ĆWICZENIA	12
4. IMPLEMENTACJA TESTÓW	14
4.1 TEMATY.....	14
4.2 CELE.....	14
4.3 WYNIKI	14
4.4 MATERIAŁY	14
4.4.1 Opracowanie przypadku testowego	14
4.4.2 Środowiska testowe	15
4.5 MATERIAŁY DODATKOWE:	15
4.6 ĆWICZENIA	15
5. WYKONANIE TESTÓW	17

5.1 TEMATY	17
5.2 CELE	17
5.3 WYNIKI.....	17
5.4 MATERIAŁY	17
5.4.1 Zarządzanie usterkami	17
5.4.2 Wdrożenie testów.....	18
5.4.3 Wykonanie testu.....	18
5.5 ĆWICZENIA.....	18
6. ZAMKNIĘCIE TESTÓW	19
6.1 TEMATY	19
6.2 CELE	19
6.3 WYNIKI.....	19
6.4 MATERIAŁY	19
6.5 ĆWICZENIA	20
7. MONITOROWANIE I KONTROLA TESTÓW	22
7.1 TEMATY	22
7.2 CELE	22
7.3 WYNIKI.....	22
7.4 MATERIAŁY	22
7.4.1 Utrzymanie testów.....	22
7.4.1 Techniki zarządzania testami	22
7.5 ĆWICZENIA	23
8. PROJEKT KOŃCOWY	24

**Licencja
publiczna**



Niniejsza publikacja © 2023 opracowana przez the Software Testing Consortium Partners jest na licencji Attribution-NonCommercial-NoDerivatives 4.0 International. Licencja dostępna pod adresem:
<http://creativecommons.org/licenses/by-nc-nd/4.0/>

0. Obszar „Mój pierwszy projekt”

Witamy w piątym obszarze ścieżki edukacyjnej „Testowanie zwinne”!

Piąty i ostatni **obszar „Mój pierwszy projekt”** służy pogłębieniu wiedzy na temat testowania oprogramowania.

Na zakończenie będziesz miał możliwość wykorzystać swoje umiejętności **i zastosować je w praktyce w projekcie, który przygotowaliśmy dla Ciebie**. To niezwykle doświadczenie stanowi nieocenioną okazję do uzupełnienia wiedzy praktycznej.



Powodzenia!

1. Planowanie testów

1.1 Tematy



- Strategia i planowanie testów
- Mapy i modele

1.2 Cele

Jako przyszły tester będziesz w stanie wykorzystać swoją wiedzę na temat strategii i planowania testów w praktyce w celu zdobycia i rozwijania umiejętności na potrzeby przyszłych projektów.



1.3 Wyniki



Po zakończeniu tego działu będziesz mógł:

- Zastosować strategię i planowanie testów w przykładowym projekcie.
- Zaplanować test ze wszystkimi jego komponentami.

1.4 Materiały

- Różnice między planowaniem testów a strategią testowania:
https://www.softwaretestinghelp.com/difference-between-test-plan-test-strategy-test-case-test-script-test-scenario-and-test-condition/#Test_Plan_Document (tekst)



- Procedury testowania:
<https://start-up.house/inventory/test-procedure-specification> (Tekst)
- Przewodnik na temat planowania testów:
<https://www.guru99.com/what-everybody-ought-to-know-about-test-planing.html> (tekst)
- Przykładowy szablon planu testowania oprogramowania
<https://www.softwaretestinghelp.com/test-plan-template/> (tekst)

- Dokument na temat strategii testowania:
<https://www.guru99.com/how-to-create-test-strategy-document.html> (tekst)
- Przykładowy szablon planu testowania:
<https://www.guru99.com/test-plan-for-project.html> (tekst)

1.5 Ćwiczenia

Po zapoznaniu się z materiałami spróbuj wykonać kilka ćwiczeń, aby wykorzystać nabytą wiedzę w praktyce.



Ćwiczenie 1:

W tej lekcji poznałeś główne koncepcje związane z planowaniem testów przy opracowywaniu projektów oceny jakości. Wymień te pojęcia i wyjaśnij *krótco* ich znaczenie.

2. Analiza testów

2.1 Tematy



- Specyfikacja wymagań biznesowych
- Analiza produktu: Przegląd techniczny
- Scenariusz testowy
- Warunki testowe

2.2 Cele

Jako przyszły tester będziesz w stanie stworzyć najlepsze praktyki, wykorzystując modele i mapy opracowane na potrzeby testowania oprogramowania. W ten sposób uzyskasz profesjonalną bazę do opracowywania projektów testowych.



2.3 Wyniki



Po zakończeniu tego działu będziesz mógł:

- Zrozumieć i przeanalizować badany produkt.
- Oszacować warunki i czas realizacji projektu.
- Zrozumieć znaczenie i efektywność mapy zależności.
- Podać wyjaśnienie dotyczące produktu, korzystając ze schematu UML w odniesieniu do reguły lub ścieżki w testowaniu oprogramowania.
- Stosować terminy używane w środowisku opracowywania oprogramowania.

2.4 Materiały

2.4.1 Analiza produktu

- Analiza produktu:

<https://www.qualtrics.com/experience-management/product/product-analysis/> (tekst)

- Analiza wymagań:

<https://www.guru99.com/learn-software-requirements-analysis-with-case-study.html> (tekst)



- Techniki szacowania testów:
<https://www.guru99.com/an-expert-view-on-test-estimation.html> (tekst +
wideo 3 min)
- Czym jest mapa zależności?
<https://miro.com/project-management/what-is-dependency-mapping/>
(tekst)
- Czym jest UML i instruktaż z przykładami:
<https://www.softwaretestinghelp.com/use-case-diagram-tutorial/> (tekst)
- Słownik pojęć związanych z testowaniem oprogramowania:
<https://www.softwaretestinghelp.com/software-testing-terms-complete-glossary/> (tekst)

2.4.2 Scenariusze i warunki testowe

- Scenariusze testowe a warunki testowe:
<https://www.guru99.com/test-scenario-vs-test-condition.html> (tekst)



2.5 Dodatkowe materiały

- Instruktaż schematu UML:
<https://youtu.be/zid-i7MVo7M-E?si=BlvsuKkCW6lcn013> (wideo,
13 min.)

2.6 Ćwiczenia

Po zapoznaniu się z materiałami spróbuj wykonać kilka ćwiczeń, aby wykorzystać nabytą wiedzę w praktyce.



Ćwiczenie 1:

W obszarze 2 „Moje środowisko”, sekcja „Analiza testów”, ćwiczenie 4, należało wybrać sklep internetowy i przetestować jego funkcjonalność w ramach ćwiczeń. Spróbuj teraz zastosować niektóre techniki analizy produktów do testowanej funkcji. Postępuj zgodnie z poniższymi zadaniami i zapisz swoje odpowiedzi:

- Dokonaj analizy produktu.

- Poproś o analizę.
- Oszacuj czas trwania tego zadania.
- Narysuj diagram w celu wyjaśnienia testowanej funkcji za pomocą modelu UML.
WSKAZÓWKA: możesz skorzystać z tego narzędzia, aby sporządzić diagram UML: <https://www.lucidchart.com>

Ćwiczenie 2:

Wyjaśnij w jednym zdaniu jak ważne jest korzystanie z narzędzi mapy zależności i wybierz dwie sytuacje, w których można ją wykorzystać.

Ćwiczenie 3:

Wymień różnice między scenariuszami testowymi a warunkami testowymi.

3. Projektowanie testów

3.1 Zagadnienia



- Architektura testów
- Narzędzia do zarządzania i przeprowadzania testów

3.2 Cele

Jako przyszły tester będziesz w stanie zidentyfikować i wybrać narzędzia do testowania oprogramowania w celu dokonania właściwych wyborów w odniesieniu do przyszłych projektów.



3.3 Wyniki



Po zakończeniu tego działu będziesz mógł:

- Opracować odpowiednią architekturę testową dla projektów.
- Wybrać narzędzia testowe na potrzeby opracowywania projektów i rozwiązywania problemów.

3.4 Materiały

3.4.1 Architektura testów

- Architektura testów:

<https://www.sixsentix.com/test-architecture-service> (tekst)

- Metoda badania modelu architektonicznego SOA:

<https://www.softwaretestinghelp.com/soa-testing/> (tekst)

- Kryteria akceptacji:

<https://resources.scrumalliance.org/Article/need-know-acceptance-criteria> (tekst)

- Kryteria akceptacji użytkownika wraz z przykładami:

<https://intellisoft.io/user-story-acceptance-criteria-explained-with-examples/> (tekst)



3.4.2 Narzędzia do zarządzania i przeprowadzania testów

- Czym jest pipeline ciągłego dostarczania?
<https://www.guru99.com/ci-cd-pipeline.html#2> (tekst)
- Znaczenie wykorzystania pipeline’u w procesie testowania:
<https://avoautomation.ai/benefits-of-ci-cd-integration-testing/> (tekst)
- Przykładowe narzędzia zarządzania testami:
<https://theqalead.com/tools/best-test-management-tools/> (tekst)
<https://www.guru99.com/testing-tools.html> (tekst)
- Strategia zautomatyzowanych testów:
<https://www.netguru.com/blog/test-automation-strategy-practices-and-examples> (tekst)
- Narzędzia do zautomatyzowanych testów:
<https://www.lambdatest.com/blog/automation-testing-tools/> (tekst)
- Czym jest Git i Git for zautomatyzowanych testów:
<https://www.atlassian.com/git/tutorials/what-is-git> (tekst)
- Ciągła integracja z Git:
<https://coderefinery.github.io/testing/continuous-integration/> (tekst)



3.5 Dodatkowe materiały

- Pipeline ciągłego dostarczania CI/CD:
<https://youtu.be/M4CXOocovZ4?si=A0N8fHHoV7r7c8Oa> (video 6 min.)

3.6 Ćwiczenia

Po zapoznaniu się z materiałami spróbuj wykonać kilka ćwiczeń, aby wykorzystać nabytą wiedzę w praktyce. Podaj krótkie odpowiedzi.



Ćwiczenie 1:

Wyjaśnij własnymi słowami, jak ważne jest tworzenie architektury w projektach testów.

Ćwiczenie 2:

Jakie są korzyści płynące z wykorzystywania pipeline'ów ciągłego dostarczania w procesach testowania oprogramowania?

Ćwiczenie 3:

Omów korzyści płynące ze stosowania Git w procesie testowania.

4. Implementacja testów

4.1 Tematy



- Opracowanie przypadku testowego
- Środowiska testowe
- Środowisko testowe konfiguracji, w tym infrastruktura.

4.2 Cele

Jako przyszły tester będziesz w stanie określić wymagania i konfiguracje w celu przeprowadzenia udanych testów przyszłych projektów.



4.3 Wyniki



Po zakończeniu tego działu będziesz mógł:

- Opracować przypadki testowe.
- Określić wymagania dotyczące środowiska testowego w swoim projekcie.
- Skonfigurować środowisko testowe w zależności od projektu, nad którym pracujesz.
- Stosować procedury testowe.

4.4 Materiały

4.4.1 Opracowanie przypadku testowego



- Czym są przypadki testowe i jak je opracować:
<https://www.browserstack.com/guide/how-to-write-test-cases> (tekst)
- Jak opracować przypadki testowe: Przewodnik na temat zapewniania jakości krok po kroku:
<https://www.coursera.org/articles/how-to-write-test-cases> (tekst)
- Planowanie testów z użyciem Jiry:
<https://www.atlassian.com/devops/testing-tutorials/jira-xray-integration-manage-test-cases> (tekst)
- Planowanie testów z użyciem TestRail:

<https://www.testrail.com/videos/introduction-projects/> (Tekst + video)

4.4.2 Środowiska testowe

- Środowisko testowe:

<https://testsigma.com/blog/test-environment/> (tekst)

- Środowisko konfiguracji testowej:

<https://www.geeksforgeeks.org/test-environment-for-software-testing/>
(tekst)

- STLC – Konfiguracja środowiska testowego

https://www.tutorialspoint.com/stlc/stlc_test_environment_setup.htm
(Tekst)

- Infrastruktura konfiguracji:

<https://www.lambdatest.com/learning-hub/test-infrastructure>
(Tekst)

- Czym jest testowanie konfiguracji i dlaczego jest ona niezbędna:

<https://www.softwaretestinghelp.com/infrastructure-testing-tutorial/>
(Tekst)



4.5 Materiały dodatkowe:

- Instrukcja na temat JIRA Xray:

<https://youtu.be/eD-miwJEL9s?si=kLa9Ggm2PRd1M4iW>

(Video 7,30 min.)

- Instrukcja TestRail:

<https://www.youtube.com/watch?v=b3yj-RCWs5I> (video 4 min.)

4.6 Ćwiczenia

Po zapoznaniu się z materiałami spróbuj wykonać kilka ćwiczeń, aby wykorzystać nabytą wiedzę w praktyce. Podaj krótkie odpowiedzi.



Ćwiczenie 1:

Wymień aspekty, które należy wziąć pod uwagę przy konfiguracji środowiska testowego. *Udziel krótkiej odpowiedzi.*

Ćwiczenie 2:

W kilku słowach wyjaśnij zalety prawidłowego utworzenia infrastruktury testowej i podaj przykłady dobrych konfiguracji.

5. Wykonanie testów

5.1 Tematy



- Zarządzanie usterkami
- Implementacja testów
- Wykonanie testów

5.2 Cele

Jako przyszły tester będziesz w stanie wdrożyć, wykonać i zidentyfikować możliwe zagrożenia w środowisku testowym.



5.3 Wyniki



Po zakończeniu tego działu będziesz mógł:

- Skutecznie wdrażać i skutecznie przeprowadzać przypadki testowe.
- Stosować techniki zarządzania możliwym ryzykiem w procesach testowych.

5.4 Materiały

5.4.1 Zarządzanie usterkami



- Kontrola jakości:

<https://www.guru99.com/defect-management-process.html> (tekst)

- Zarządzanie usterkami:

<https://www.lambdatest.com/learning-hub/defect-management> (tekst)

- Sprawozdanie z błędów:

<https://www.guru99.com/defect-management-process.html> (tekst)

- Technika identyfikacji ryzyka:

<https://icepanel.medium.com/risk-storming-31e89b0ac42> (tekst)

- Strategia techniki identyfikacji ryzyka ze sferą testową:

<https://www.ministryoftesting.com/testsphere/riskstorming> (tekst + video, 3 min.)

5.4.2 Wdrożenie testów

- Implementacja testów:
<https://blog.santexgroup.com/blog/7-steps-for-software-implementation-success/> (tekst)
- Jak stworzyć plan wdrażania oprogramowania:
<https://technologyadvice.com/blog/information-technology/software-implementation-timeline/> (tekst)

5.4.3 Wykonanie testu

- Przeprowadzenie testów:
<https://www.softwaretestinghelp.com/test-execution-software-testing-qa-training-on-a-live-project-day-5/> (tekst)
- Instrukcja na temat przeprowadzania testów:
<https://www.lambdatest.com/learning-hub/test-execution> (tekst)
- Test integracji systemów:
<https://www.guru99.com/system-integration-testing.html> (tekst)

5.5 Ćwiczenia

Po zapoznaniu się z materiałami spróbuj wykonać kilka ćwiczeń, aby wykorzystać nabytą wiedzę w praktyce. Podaj krótkie odpowiedzi.



Ćwiczenie 1:

Podaj zalety i znaczenie dobrego zarządzania usterkami. Wymień przykładowe techniki zarządzania usterkami. *Udziel krótkich odpowiedzi.*

Ćwiczenie 2:

Wymień niezbędne kroki, jakie należy podjąć w celu pomyślnego wdrożenia oprogramowania.

Ćwiczenie 3:

Wymień niektóre z działań podejmowanych podczas wykonywania testów.

6. Zamknięcie testów

6.1 Tematy



- Raport z wykonania testów
- Wyniki testów
- Kryteria wyjścia: definicja „wykonane” w procesie testowania

6.2 Cele

Jako przyszły tester będziesz w stanie wykorzystać swoją wiedzę w celu dalszego jej przekazania, aby usprawnić pracę w zespole w przyszłych zespołach projektowych.



6.3 Wyniki



Po zakończeniu tego działu będziesz mógł:

- Stworzyć szablon listy kontrolnej na zamknięcie testów.
- Zrozumieć, jak ważne jest dokumentowanie cyklu życia w procesie testowania.
- Zdefiniować pojęcie „wykonane”.

6.4 Materiały

- Dokumentacja testowa:

<https://strongqa.com/qa-portal/knowledge-base/key-concepts/testing-documentation> (tekst)

- Przykłady szablonów testowych:

<https://marker.io/blog/usability-testing-template> (tekst)

- Archiwizacja wyników testów:

<https://cxl.com/blog/archiving-test-results/> (tekst)

- Archiwizacja przypadków testowych:

<https://qmetrysupport.atlassian.net/wiki/spaces/QTM4/pages/2582872829/Archive+Test+Cases> (tekst)



- Kryteria akceptacji/definicja pojęcia ‘wykonane’:
<https://www.mindfulqa.com/acceptance-criteria/> (tekst)

6.5 Ćwiczenia

Po zapoznaniu się z materiałami spróbuj wykonać kilka ćwiczeń, aby wykorzystać nabytą wiedzę w praktyce.



Ćwiczenie 1:

Wymień co najmniej 4 etapy procesu dokumentowania testów i wyjaśnij znaczenie każdego z nich. Podaj zwięzłe odpowiedzi (maksymalnie dwa zdania na każdy etap).

Ćwiczenie 2:

Dlaczego archiwizacja przypadków testowych ma zasadnicze znaczenie w procesie testowania?

Ćwiczenie 3:

Kiedy można zastosować definicję „wykonane” w odniesieniu do projektu?

Zadanie 4:

Teraz pokrótce przeanalizujemy dotychczasowe postępy w nauce. Pomyśl o każdym etapie projektu testowego (planowanie, analiza, projektowanie, implementacja, wykonanie, zamknięcie oraz monitorowanie/kontrola) i zapisz odpowiedzi na następujące pytania:

- W jaki sposób przebiegał dotychczasowy cały proces edukacyjny?
- Czy jest coś co budzi twoje wątpliwości? Co do tej pory było dla Ciebie najtrudniejsze?
- Czy uważasz, że musisz dopracować jakiś konkretny aspekt, aby stać się dobrym testerem w zakresie zapewniania jakości? Czy jest jakiś konkretny temat, o którym chciałbyś się więcej dowiedzieć?

7. Monitorowanie i kontrola testów

7.1 Tematy



- Utrzymanie testowania oprogramowania

7.2 Cele

Jako przyszły tester będziesz w stanie dokonać sprawdzenia całego procesu testowania i w ten sposób zapewnić utrzymanie projektu, promując w razie potrzeby nowe zmiany, zachowując integralność i dobrą jakość produktu.



7.3 Wyniki



Po zakończeniu tego działu będziesz mógł:

- Opracować techniki utrzymania testów.

7.4 Materiały

7.4.1 Utrzymanie testów



- Praktyki utrzymania:

<https://www.softwaretestinghelp.com/test-monitoring-and-test-control/>
(tekst)

- Co to jest konserwacja oprogramowania i dlaczego jest ona tak istotna:

<https://radixweb.com/blog/why-software-maintenance-is-necessary#Maintenance> (tekst)

7.4.1 Techniki zarządzania testami

- Jak skutecznie zarządzać usterką:

<https://www.geeksforgeeks.org/defect-management-process/>
(Tekst)

- Przewodnik na temat skutecznego zarządzania usterkami w oprogramowaniu:

<https://www.kualitee.com/defect-management-tool/step-by-step-guide-to-effective-software-defect-management/> (tekst)

7.5 Ćwiczenia

Po zapoznaniu się z materiałami spróbuj wykonać kilka ćwiczeń, aby wykorzystać nabytą wiedzę w praktyce.



Ćwiczenie 1:

Wymień procesy, które są częścią konserwacji oprogramowania.

8. Projekt końcowy

Wykorzystamy teraz zdobytą wiedzę w praktyce w ramach ścieżki edukacyjnej „Testowanie zwinne”. W ramach niniejszego projektu końcowego zostaniesz poproszony o zastosowanie technik, z którymi zapoznałeś się we wszystkich obszarach. Należy **wziąć pod uwagę techniki i narzędzia istotne dla wszystkich kluczowych etapów testowania**, takie



jak:

- Planowanie testów
- Analiza testów
- Projektowanie testów
- Implementacja testów
- Wykonanie testów
- Zamknięcie testów
- Monitorowanie i kontrola testów

Pamiętaj, że w razie jakichkolwiek wątpliwości lub trudności **możesz zawsze sięgnąć do materiałów szkoleniowych dostępnych we wszystkich obszarach w celu uzyskania jak najlepszej odpowiedzi na to zadanie.**

Po udzieleniu odpowiedzi w projekcie omów je się ze swoim trenerem.

Powodzenia!

TWÓJ PIERWSZY PROJEKT

Projekt będzie koncentrował się na przetestowaniu bardzo istotnej funkcji niezbędnej dla każdej formy handlu elektronicznego: **nabywanie rzeczy**.

- Otwórz stronę internetową <https://www.renfe.com/es/en>
- Funkcja do przetestowania: *zakup biletu*.
- Warunki testowe: Klient musi być zalogowany na platformie.
- Uwaga: W części dotyczącej płatności biletu nie będziesz mógł kontynuować działania, ale możesz opisać, co należy zrobić i jakie techniki testowania należy zastosować w celu pełnego przetestowania funkcji.

Następnie zaproponuj wszystkie działania niezbędne do przetestowania funkcji z uwzględnieniem wszystkich faz testów oraz powiązanych narzędzi i technik. Podaj wszystkie możliwe szczegóły.