

**TESTUL Nr.2**

- 1 Enumerati avantajele si dezavantajele modelului ciclului de viata Cascada (Waterfall) a produsului software. 4p.

<i>Avantaje</i>	<i>Dezavantaje</i>
<ol style="list-style-type: none"><li>1. Simplu și ușor de înțeles și de utilizat</li><li>2. Ușor de gestionat datorită rigidității modelului. Fiecare fază are rezultate specifice și un proces de revizuire.</li><li>3. Etape clar definite.</li><li>4. Funcționează bine pentru proiecte mai mici, unde cerințele sunt simple foarte ușor înțelese.</li><li>5. Repere/ etape bine înțelese.</li></ol>	<ol style="list-style-type: none"><li>1. Niciun software de lucru nu este produs până în etapele de sfârșit ciclului de viață.</li><li>2. Niveluri ridicate de risc și incertitudine.</li><li>3. Este dificil să se măsoare progresele înregistrate în cadrul etapelor.</li><li>4. Nu este un model bun pentru proiecte complexe și orientate pe obiecte.</li><li>5. Este complicat adapta cerințelor în schimbare.</li></ol>

2. Care este diferența dintre Severitatea (Severity) și Prioritatea (Priority) defectului? Da-ti exemple. 4p.

<i>Priority</i>	<i>Severity</i>
<p>Este definit ordinea în care dezvoltatorul ar trebui să rezolve un defect</p> <p>Este clasificată în trei tipuri:</p> <ul style="list-style-type: none"><li>➤ Low</li><li>➤ Medium</li><li>➤ High</li></ul> <p>Indică cât de curând trebuie remediată eroarea</p> <p>Prioritatea defectelor este decisă în consultare cu managerul / clientul</p> <p>Statutul de prioritate se bazează pe cerințele clienților</p>	<p>Este definită ca gradul de impact pe care un defect îl are asupra funcționării produsului</p> <p>Este clasificată în cinci tipuri:</p> <ul style="list-style-type: none"><li>➤ Critical</li><li>➤ Major</li><li>➤ Moderate</li><li>➤ Minor</li><li>➤ Cosmetic</li></ul> <p>Indică gravitatea defectului asupra funcționalității produsului</p> <p>Persoana ce se ocupa de QA determină gradul de severitate al defectului</p> <p>Starea de severitate se bazează pe aspectul tehnic al produsului</p>

## 3 Tehnici de testare. White Box, Grey Box, Black Box – comparații-le

7p.

<i>White Box</i>	<i>Grey Box</i>	<i>Black Box</i>
Cunoașterea structurii interne de lucru (codificarea software-ului) este necesară în mod necesar pentru acest tip de testare.	Este necesară cunoașterea parțială a structurii interne de lucru.	Cunoașterea structurii interne de lucru (cod) nu este necesară pentru acest tip de testare. Numai GUI (Graphical User Interface) este necesar pentru cazurile de testare.
Se efectuează prin verificarea limitelor sistemului și a domeniilor de date inerente software-ului, deoarece nu lipsesc cunoștințele interne de codificare.	Dacă testerul are cunoștințe de codificare, atunci se procedează prin validarea domeniilor de date și a limitelor sistemului intern al software-ului.	Abordarea față de testare include <i>trial techniques</i> și <i>error guessing method</i> deoarece testerul nu are nevoie de cunoștințe de cod interne a software-ului.
Spațiul de testare a tabelor pentru intrări (intrări care urmează să fie utilizate pentru crearea cazurilor de testare) este mai mic în comparație cu <i>Black Box Testing</i> .	Spațiul de testare a tabelor pentru intrări (intrări care trebuie utilizate pentru crearea cazurilor de testare) este mai mic decât la <i>White Box &amp; Black Box Testing</i> .	Spațiul de testare a tabelor pentru intrări (intrări care trebuie utilizate pentru crearea cazurilor de testare) este destul de imens și cel mai mare dintre toate tipurile de testare (White/ Grey Box Box Testing).
Durează mult timp pentru a proiecta cazuri de testare din cauza codului lung.	Proiectarea cazurilor de testare se poate face într-o perioadă scurtă de timp.	Consumul de timp în <i>Black Box Testing</i> depinde de disponibilitatea specificațiilor funcționale.
Tester-ul și Developer-ul fac parte în procesul de testare.	Tester-ul, Developer-ul și utilizatorul final poate face parte în procesul de testare.	Tester-ul, Developer-ul și utilizatorul final poate face parte în procesul de testare.
Rezistența și securitatea împotriva atacurilor de virusi ( <i>viral attacks</i> ) nu sunt acoperite în cadrul acestui tip de test.	Rezistența și securitatea împotriva atacurilor de virusi ( <i>viral attacks</i> ) nu sunt acoperite în cadrul acestui tip de test.	Rezistența și securitatea împotriva atacurilor de virusi ( <i>viral attacks</i> ) sunt acoperite în cadrul acestui tip de test.
Consuma/ Necesita cel mai mult timp pentru a fi efectuat	Consuma/ Necesita mai puțin timp ca <i>White Box Testing</i>	Consuma/ Necesita cel mai puțin timp pentru a fi efectuat

4. Utilizatorul trebuie să aibă posibilitatea de a căuta zboruri pe pagina Homepage>Flightschedule>Flight search. Rezultatul căutării va fi afișat după completarea datelor pentru câmpuri: Destination, Departure, Arrival. 10p.

În timpul executării testului, ați identificat un defect: rezultatele căutării arată datele zborurilor din trecut. Elaborați raportul despre acest defect (*defect report*), specificând atributele de bază pentru descrierea defectului.

<b>ID</b>	A404
<b>PROJECT</b>	Flight America – Flight scheduling
<b>PRODUCT</b>	International Flight Scheduler
<b>RELEASE VERSION</b>	1.4.0a
<b>MODULE</b>	Homepage>Flightschedule>Flight
<b>DETECTED BUILD VERSION</b>	1.4.0.3a
<b>SUMMARY</b>	As a result it was displaying results dated in past, not in the time range indicated in the fields
<b>DESCRIPTION</b>	While running several defect tests on finding flight schedules completing the <i>Destination</i> , <i>Departure</i> , <i>Arrival</i> , with different data, multiple results were displayed dated in past, not in the time range indicated (Departure - Arrival).
<b>STEPS TO REPLICATE</b>	1.1 "Choose destination" → Moscow 1.2 "Departure" → 13:30 30.11.2021 1.3 "Arrival" → 21:30 30.11.2021  2.1 "Choose destination" → London 2.2 "Departure" → 11:30 30.11.2021 2.3 "Arrival" → 00:30 1.12.2021  3.1 "Choose destination" → Zurich 3.2 "Departure" → 15:00 30.11.2021 3.3 "Choose destination" → 20:30 30.11.2021
<b>ACTUAL RESULT</b>	<i>To: Moscow 17:30 28.11.2021 22:30 28.11.2021</i>  <i>To: London 11:30 28.11.2021 00:30 29.11.2021</i>  <i>To: Zurich 15:30 28.11.2021 21:00 28.11.2021</i>
<b>EXPECTED RESULT</b>	<i>To: Moscow 13:30 30.11.2021 21:30 23.11.2021</i>  <i>To: London 11:30 30.11.2021 00:30 01.12.2021</i>  <i>To: Zurich 15:00 30.11.2021 20:30 30.11.2021</i>

<b>ATTACHMENT</b>	<i>[SCREENSHOT]* [SCREENSHOT]</i>
<b>REMARKS</b>	The problem seems to be in the JS code which is responsible of filtering the flight schedules
<b>DEFECT PROBABILITY</b>	HIGH
<b>DEFECT SEVERITY</b>	MAJOR
<b>DEFECT PRIORITY</b>	HIGH
<b>REPORTED BY</b>	Fernanda BARRERO
<b>ASSIGNED TO</b>	Catalin PAVLOVSKI
<b>STATUS</b>	ASSIGNED
<b>FIXED BUILD VERSION</b>	1.4.0.5a

\*[TAG\_NAME] – este un tag pt substituire a imaginilor, semnaturilor, logorilor in caz ca acestea nu pot fi recreate

Nota	„10”	„9”	„8”	„7”	„6”	„5”	„4”	„3”	„2”	„1”
Nr. de puncte	25-24	23-22	21-19	18-16	15-11	10-7	6-5	4-3	2	1