# UC4 – Συνδρομητική Υπηρεσία – White Box Testing

## 1. Κατασκευή Γράφου Ροής Ελέγχου (Flow Graph)

## def validate\_and\_submit(self): # 1: Αρχή συνάρτησης

## valid = True

## for entry, label in [

## (self.card\_number\_entry, self.card\_number\_error),

## (self.expiry\_entry, self.expiry\_error),

## (self.card\_name\_entry, self.card\_name\_error)

## ]:

## entry.config(highlightbackground="gray", highlightthickness=1)

## label.config(text="")

## card\_number = self.card\_number\_entry.get()

## if not card\_number.isdigit(): # 2: Έλεγχος αν αριθμός κάρτας είναι ψηφιακός

## self.card\_number\_entry.config(highlightbackground="red", highlightthickness=2)

## self.card\_number\_error.config(text="Μόνο αριθμοί επιτρέπονται") # 3: Μήνυμα σφάλματος

## valid = False

## elif len(card\_number) != 16: # 4: Έλεγχος αν έχει μήκος 16 ψηφία

## self.card\_number\_entry.config(highlightbackground="red", highlightthickness=2)

## self.card\_number\_error.config(text="Ο αριθμός πρέπει να περιέχει 16 ψηφία") # 5: Μήνυμα σφάλματος

## valid = False

## expiry = self.expiry\_var.get()

## if len(expiry) != 5 or expiry[2] != "/" or not expiry.replace("/", "").isdigit(): # 6: Έλεγχος μορφής λήξης

## self.expiry\_entry.config(highlightbackground="red", highlightthickness=2)

## self.expiry\_error.config(text="Μορφή: MM/YY") # 7: Μήνυμα σφάλματος

## valid = False

## card\_name = self.card\_name\_entry.get()

## if not all(ord(c) < 128 for c in card\_name): # 8: Έλεγχος λατινικών χαρακτήρων

## self.card\_name\_entry.config(highlightbackground="red", highlightthickness=2)

## self.card\_name\_error.config(text="Μόνο λατινικοί χαρακτήρες") # 9: Μήνυμα σφάλματος

## valid = False

## if valid:

## if self.db.userHasActiveSubscription(CURRENT\_USER\_ID): # 10: Έλεγχος ενεργής συνδρομής

## messagebox.showinfo("Πληροφορία", "Έχετε ήδη ενεργή συνδρομή.") # 11: Μήνυμα

## else:

## self.db.insertSubscription(CURRENT\_USER\_ID, self.subscriptionType) # 12: Εισαγωγή συνδρομής

## self.window.destroy() # 13: Τέλος συνάρτησης

## 

## 2. Πίνακας Καταστάσεων Κόμβων (Nodes Table)

|  |  |
| --- | --- |
| Κόμβος | Περιγραφή |
| 1 | Αρχικοποίηση μεταβλητών, καθαρισμός προηγούμενων σφαλμάτων |
| 2 | Έλεγχος αν αριθμός κάρτας περιέχει μόνο ψηφία |
| 3 | Εμφάνιση μηνύματος σφάλματος (όχι ψηφία) |
| 4 | Έλεγχος μήκους αριθμού κάρτας |
| 5 | Εμφάνιση μηνύματος σφάλματος (όχι 16 ψηφία) |
| 6 | Έλεγχος μορφής ημερομηνίας λήξης |
| 7 | Εμφάνιση μηνύματος σφάλματος (λάθος μορφή) |
| 8 | Έλεγχος αν όνομα κατόχου είναι λατινικό |
| 9 | Εμφάνιση μηνύματος σφάλματος (μη λατινικό) |
| 10 | Έλεγχος αν υπάρχει ενεργή συνδρομή |
| 11 | Εμφάνιση μηνύματος 'έχετε ήδη ενεργή συνδρομή' |
| 12 | Καταχώρηση συνδρομής και πληρωμής στη βάση |
| 13 | Κλείσιμο παραθύρου |

## 3. Πίνακας Ακμών Ελέγχου (Edges Table)

|  |  |
| --- | --- |
| Από | Προς |
| 1 | 2 |
| 2 | 3 (αν False) |
| 2 | 4 (αν True) |
| 4 | 5 (αν False) |
| 4 | 6 (αν True) |
| 6 | 7 (αν False) |
| 6 | 8 (αν True) |
| 8 | 9 (αν False) |
| 8 | 10 (αν True) |
| 10 | 11 (αν True) |
| 10 | 12 (αν False) |
| 11 | 13 |
| 12 | 13 |

## 4. Προσδιορισμός Βασικών Μονοπατιών (Independent Paths)

* Μονοπάτι 1: 1 → 2 (False) → 3 → 13
* Μονοπάτι 2: 1 → 2 (True) → 4 (False) → 5 → 13
* Μονοπάτι 3: 1 → 2 → 4 → 6 (False) → 7 → 13
* Μονοπάτι 4: 1 → 2 → 4 → 6 → 8 (False) → 9 → 13
* Μονοπάτι 5: 1 → 2 → 4 → 6 → 8 → 10 (True) → 11 → 13
* Μονοπάτι 6: 1 → 2 → 4 → 6 → 8 → 10 (False) → 12 → 13

## 5. Αντιστοίχιση με Test Cases (Test Case Mapping)

|  |  |  |
| --- | --- | --- |
| Test Case | Διαδρομή | Περιγραφή |
| TC1 | 1-2-3-13 | Λανθασμένος χαρακτήρας στον αριθμό κάρτας |
| TC2 | 1-2-4-5-13 | Ορθός τύπος, αλλά λιγότερα/περισσότερα από 16 ψηφία |
| TC3 | 1-2-4-6-7-13 | Μη έγκυρη ημερομηνία λήξης (format λάθος) |
| TC4 | 1-2-4-6-8-9-13 | Μη λατινικοί χαρακτήρες στο όνομα κατόχου |
| TC5 | 1-2-4-6-8-10-11-13 | Έγκυρα δεδομένα αλλά υπάρχει ήδη ενεργή συνδρομή |
| TC6 | 1-2-4-6-8-10-12-13 | Έγκυρα δεδομένα, νέα εγγραφή |

A screen shot of a computer

AI-generated content may be incorrect.