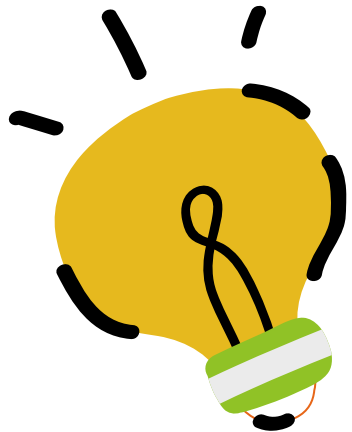


Proiect Final
Rădeanu Lucian-Ștefan
08.04.2023



Explicați ce sunt acelea requirements

- ❖ **Requirements** sunt cerințele sau specificațiile funcționale și non-funcționale ale unui sistem sau produs software, care trebuie să fie îndeplinite pentru a satisface nevoile utilizatorilor sau ale business-ului.

Explicați ce este un test condition

- ❖ **Test Condition:** Un element al unei componente sau al unui sistem care poate fi verificat printr-un test case, cum ar fi o funcție, o tranzacție, o funcționalitate, un atribut etc. Răspunde la întrebarea: Ce voi testa?

Explicați ce este un test case și la ce folosește

- ❖ **Test Case:** Un set de input-uri, condiții de execuție (precondiții), pași de execuție și rezultate așteptate descrise în raport cu un obiectiv bine specificat, de obicei definit într-un test condition. Testele de caz sunt folosite pentru a evalua dacă un sistem sau produs software îndeplinește cerințele și specificațiile funcționale și non-funcționale. Răspunde la întrebarea: Cum voi testa?



Explicați cum se poate folosi un test plan (test suite) care este scopul unui test plan

- ❖ Un **test plan** este un document care descrie în detaliu cum vor fi testate anumite caracteristici sau funcționalități ale unui produs sau sistem software.



- ❖ **Scopul principal** al unui test plan este de a asigura că toate aspectele importante ale produsului sau sistemului sunt testate și verificate, astfel încât să se poată detecta și remedia eventualele erori sau probleme înainte ca produsul sau sistemul să fie livrat clienților sau utilizatorilor finali.

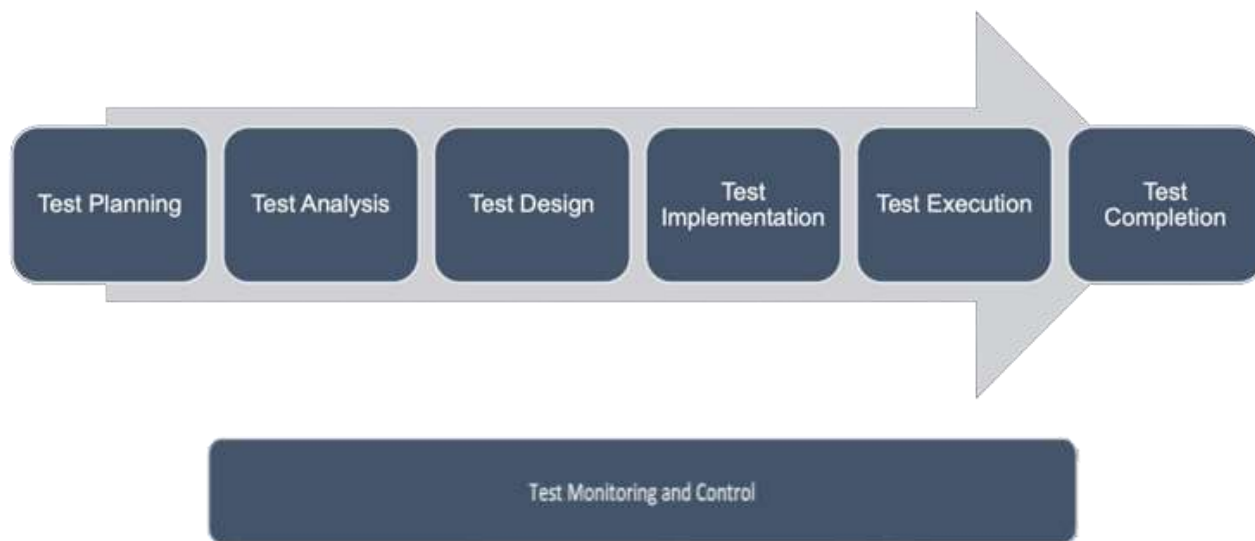
Enumerați statusurile pe care poate să le aibă rularea unui test case.

- ❖ **Passed** (trecut) - testul a fost executat cu succes și a obținut rezultatele așteptate.
- ❖ **Failed** (eșuat) - testul a fost executat și nu a obținut rezultatele așteptate. Acest lucru poate fi cauzat de o problemă în software-ul testat sau de o problemă în testul însuși.
- ❖ **Blocked** (blocat) - testul nu a putut fi executat din cauza unei probleme tehnice, cum ar fi o dependență care lipsește sau un alt test care trebuie să fie efectuat înainte.
- ❖ **In progress** (în desfășurare) - testul este încă în procesul de a fi executat și nu a fost încă finalizat.
- ❖ **Skipped** (sărit) - testul a fost omis din motive variate, cum ar fi depășirea timpului alocat sau faptul că testul nu este relevant pentru acea versiune de software.
- ❖ **Pending** (în așteptare) - testul nu a fost încă efectuat și este așteptat să fie efectuat în viitor.
- ❖ **Retest** (retestare) - testul a fost eșuat, dar a fost corectat sau modificat software-ul și acum trebuie să fie retestat.
- ❖ **Not applicable** (nu este aplicabil) - testul nu este relevant pentru acea versiune de software sau pentru anumite condiții de testare.

Enumerați statusurile pe care poate să le aibă un defect

- ❖ **New** (nou) - defectul a fost identificat și raportat pentru prima dată și încă nu a fost evaluat.
- ❖ **Assigned** (atribuit) - defectul a fost evaluat și atribuit unui dezvoltator sau altui membru al echipei pentru a fi remediat.
- ❖ **In progress** (în desfășurare) - dezvoltatorul sau alt membru al echipei lucrează la remedierea defectului.
- ❖ **Fixed** (remediat) - dezvoltatorul sau alt membru al echipei a remediat defectul și acesta a fost verificat.
- ❖ **Rejected** (respins) - defectul a fost evaluat și considerat a fi un comportament așteptat, astfel încât nu este necesară remedierea.
- ❖ **Deferred** (amânat) - remedierea defectului a fost amânată pentru o versiune viitoare a software-ului.
- ❖ **Duplicate** (duplicat) - defectul este duplicat al altui defect existent și raportat deja.
- ❖ **Cannot Reproduce** (nu se poate reproduce) - defectul nu poate fi reproducibil, deci nu poate fi verificat și remediat.
- ❖ **Closed** (închis) - defectul a fost remediat și verificat cu succes și a fost închis.
- ❖ **Reopened** (re-deschis) - defectul a fost închis, dar a reapărut ulterior și trebuie remediat din nou.

Explicați care sunt etapele procesului de testare



- ❖ **Planificare:** definirea obiectivelor, specificarea cerințelor și planificarea testelor.
- ❖ **Analiză:** Răspunde la întrebarea: “Ce urmează să testăm?”. În această etapă se analizează documentația primită de la client pentru a ne asigura că le înțelegem, de asemenea, în această etapă se generează condițiile de testare (Ce vom testa)
- ❖ **Design:** Răspunde la întrebarea : “Cum vom testa?”. În aceasta etapă se creează cazurile de testare și se prioritizează. De asemenea se identifică datele de testare de care avem nevoie și se face design-ul mediului de testare prin identificarea oricărui tool sau infrastructură de care avem nevoie pentru testare.

- ❖ **Implementare:** Răspunde la întrebarea : “Avem tot ce ne trebuie pentru a începe executarea testelor?” În această etapă se creează datele de testare identificate în etapa anterioară. Se creează mediul de test și tot în această etapă ne asigurăm că a fost definit în mod corect. Testele se prioritizează, se grupează pe baza obiectivelor lor. Tot în etapa aceasta ne asigurăm că avem tot ce ne trebuie pentru a începe testarea propriu-zisă.
- ❖ **Execuție:** În această etapă cazurile de testare sunt executate. Rezultatele sunt raportate în tool-ul în care au fost scrise testele. Bug-urile sunt raportate atunci când rezultatele așteptate nu coincid cu rezultatele actuale. Atunci când bug-urile sunt fixate, se face retestarea lor pentru a ne asigura că au fost într-adevăr fixate.
- ❖ **Închidere:** În această etapă se evaluează criteriile de ieșire pentru a ne asigura că putem să închidem procesul de testare în siguranță. Orice task-uri rămase deschise și bug-uri sunt re-evaluate și ulterior închise. Materialele de testare sunt predate și arhivate (handover of testware). Este generat un raport de închidere a testării care ulterior este trimis către stakeholders.
- ❖ **Monitorizare și Control:** Este o activitate continuă care se desfășoară cu scopul de a compara progresul actual cu planul de testare (monitorizare). Începe o dată cu etapa de planificare și se termină o dată cu etapa de închidere. Presupune raportare periodică prin rapoarte de status care să includă și notificarea oricărui devieri de la plan, în cazul în care se observă riscul de a nu ne îndeplini obiectivele, se iau măsuri de control.

Explicati diferenta intre retesting și regression testing

- ❖ **Retesting** este un proces prin care se verifica daca defectele marcate ca remediate au fost intradevar remediate, iar **regression testing** este un proces prin care se verifica programul sau o parte din program pentru a ne asigura ca schimbarile aduse asupra lui nu au cauzat/descoperit alte defecte



Explicati diferenta intre functional testing si non-functional testing

- ❖ Functional testing se referă la testarea funcționalității aplicației și verificarea dacă aceasta îndeplinește cerințele de afaceri și funcționale. Aceasta se concentrează pe testarea comportamentului și a caracteristicilor aplicației
- ❖ Non-functional testing se referă la testarea caracteristicilor non-funcționale ale aplicației, cum ar fi performanța, securitatea, scalabilitatea, fiabilitatea etc. Scopul este de a asigura că aplicația este adecvată din punct de vedere tehnic și poate funcționa corect în condiții variate.



Enumerati tehnicile de testare și grupati-le în categoria corespunzătoare (Black-Box, White-Box, Experience-Based)

Experience-Based Testing:

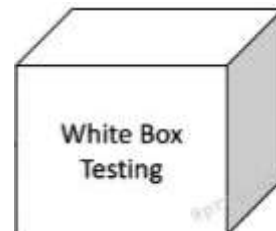
- ❖ Testare exploratorie
- ❖ Smoke Testing
- ❖ Sanity Testing

White-Box Testing:

- ❖ Testare exploratorie
- ❖ Decision Coverage
- ❖ Complexitatea Ciclomatică

Black-Box Testing:

- ❖ Partiționarea Echivalentă.
- ❖ Analiza Valorilor de Graniță
- ❖ Testarea Tranzițiilor de State
- ❖ Tabel Decizional



Am ales să prezint o aplicație delivery, și anume aplicația de livrare a mâncării, Glovo.

Cerințele pentru aplicația Glovo:



1. Să permită utilizatorilor să plaseze comenzi pentru livrarea de mâncare de la restaurante partenere.
2. Să permită utilizatorilor să filtreze opțiunile de livrare după locație, timpul de livrare și opțiuni de plată.
3. Să ofere opțiunea de a vizualiza meniurile restaurantelor partenere și de a adăuga și elimina articole din coșul de cumpărături.
4. Să permită utilizatorilor să urmărească starea comenzilor lor și să primească actualizări despre acestea.
5. Să ofere posibilitatea de a evalua și a lăsa recenzii pentru restaurantele partenere și pentru șoferii de livrare.
6. Să permită utilizatorilor să plătească pentru comanda lor prin intermediul aplicației, utilizând opțiuni de plată precum cardurile bancare sau portofelele electronice.

Detalii pentru cerințele identificate:

1. Utilizatorii trebuie să poată selecta un restaurant și să aleagă articolele pe care doresc să le comande.
2. Utilizatorii trebuie să poată introduce o adresă de livrare și să filtreze opțiunile de livrare după timpul de livrare și opțiunile de plată disponibile.
3. Aplicația trebuie să afișeze meniurile restaurantelor partenere și să permită utilizatorilor să adauge și să elimine articole din coșul de cumpărături.
4. Utilizatorii trebuie să poată urmări starea comenzilor lor și să primească actualizări despre acestea, inclusiv confirmarea comenzii, informații despre preluarea și livrarea comenzii și informații despre șoferul de livrare.
5. Utilizatorii trebuie să poată evalua și lăsa recenzii pentru restaurantele partenere și pentru șoferii de livrare după finalizarea comenzii lor.
6. Utilizatorii trebuie să poată plăti pentru comanda lor prin intermediul aplicației, utilizând opțiuni de plată precum cardurile bancare sau portofelele electronice.

Raportul general în aplicația Jira

GlovoApp

Projects /  Lucian /  RLS-14



Test Cases

Înregistrarea ca și livrator

 Attach  Create subtask  Link issue  Zephyr Actions  Test Details  Test Executions ...

Description

- Deschide aplicația Glovo și selectează opțiunea "Înregistrează-te ca și livrator".
- Completează detaliile tale personale, inclusiv numele, adresa, numărul de telefon, școala la care ai învățat și alte informații relevante.
- Încarcă toate documentele necesare pentru a deveni un livrator Glovo (carte de identitate, permis de conducere, certificat de înregistrare a vehiculului, etc.).
- Așteaptă confirmarea din partea echipei Glovo cu privire la acceptarea ta ca livrator.
- Începe să primești comenzi și să le livrezi către clienți, folosind aplicația Glovo pentru a gestiona comenzile și locațiile.
- Asigură-te că respecti cerințele de livrare (timpul estimat, calitatea produselor, adresa de livrare, etc.).
- Închide comanda în aplicație și așteaptă feedback-ul clientului în legătură cu livrarea.
- Continuă să livrezi comenzile și să primești feedback pentru a îmbunătăți experiența clienților.

Projects /  Lucian /  RLS-13

Plasarea unei comenzi de livrare

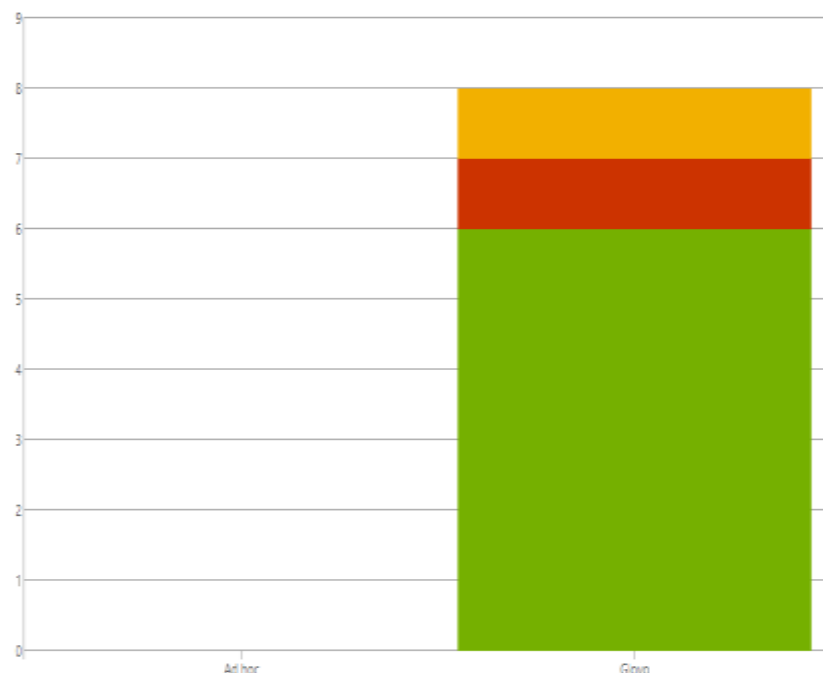
 Attach  Create subtask  Link issue  Zephyr Actions  Test Details  Test Executions ...

Description

- 1.Deschide aplicația Glovo și autentifică-te în contul tău existent sau creează un cont nou.
2. În meniul principal, selectează categoria de produse pe care dorești să le comanzi (alimente, produse de curățenie, medicamente, etc.).
3. Navighează prin listă sau folosește bara de căutare pentru a găsi produsul specific pe care dorești să îl comanzi.
4. Adaugă produsul în coșul de cumpărături.
5. Continuă să adaugi alte produse, sau finalizează comanda și verifică că detaliile comenzii (produse, cantități, adresa de livrare) sunt corecte.
6. Apasă butonul "Plasează comanda".
7. Așteaptă confirmarea comenzii și estimarea timpului de livrare.
8. Urmează statusul comenzii în aplicație, de la preluarea de către curier, la pregătirea coletului și livrarea la adresa specificată.
9. Primește produsul comandat și verifică că este ceea ce ai comandat și în stare bună.

Test Execution

Lucian (RLS) / Glovo
Test Execution by cycle



 PASS  FAIL  WIP  BLOCKED  UNEXECUTED

Curierul nu poate apasa pe pin-ul de pe harta cand primeste o comanda pentru a vedea detalii

Attach Create subtask Link issue

Description

Normal text B I ... A ...

Descrie bug-ul

Utilizatorul nu poate face click pe locatia indicatorului adresei de ridicare/livrare (galben).

Pași de reproducere: Conectati-vă la aplicație cu contul de curier, când primiți comanda de ridicare de la restaurant, va afișa o hartă cu indicatorul de locație exactă pe hartă, când faceți clic pe ea, va afișa numele și adresa restaurantului și deschide linkul într-o aplicație de navigare.

Rezultatul actual: Curierul nu poate face clic pe acele puncte de locație. Nu se întâmplă nimic făcând clic.

Rezultatul așteptat: Curierul poate face clic pe acele puncte de locație și poate vedea detaliile despre comanda.

S-au adăugat capturi de ecran ale aplicației care funcționează sub serviciile Google Play

Environment

Sistem

Versiunea Android: [11]

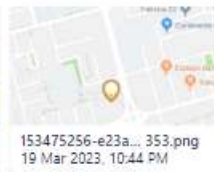
ROM personalizat: [LineageOS_onclite-userdebug rq3a.21101]

microG

Versiunea microG Core: [0.2.24.214816]

Rezultate autoverificare microG: [Toate bifate]

Attachments (2)



Raportul de defect - bug ticket

Analiza de risc

Riscuri de produs

1. Probleme tehnice.
2. Reglementări și conformitate(Riscul ca aplicația să nu fie conformă cu reglementările și legile locale sau internaționale)
3. Performanța rețelei și a serverelor
4. Securitatea datelor(riscul ca datele utilizatorilor să fie vulnerabile la atacurile cibernetice sau că acestea să fie compromise din alte motive)
5. Concurența(riscul ca aplicația să fie depășită de alte aplicații similare care oferă aceleași servicii)

Riscuri de proiect

1. Resursele insuficiente: Riscul ca proiectul să nu aibă suficiente resurse, cum ar fi timpul, bugetul sau personalul, pentru a fi finalizat în termenii stabiliți.
2. Imposibilitatea de a livra produsul

Matricea trasabilității

Cerințe	Teste de acceptare	Cod
Cerințe de utilizare	Teste de acceptare	Teste de performanță
Autentificare în contul existent sau creare de cont nou	Testarea funcționalității de autentificare	Testarea timpului de răspuns al aplicației la cererea de autentificare
Plasarea unei comenzi	Testarea adăugării de produse în coșul de cumpărături și finalizarea comenzii	Testarea timpului de procesare a comenzii
Livrarea comenzilor	Testarea funcționalității de gestionare a comenzilor și de monitorizare a stării comenzilor	Testarea timpului de livrare și de calitate a livrării
Înregistrarea ca și livrator	Testarea procesului de înregistrare și de încărcare a documentelor necesare	Testarea performanței de gestionare a comenzilor și de navigare către locații
Cerințe de utilizare	Teste de acceptare	Teste de performanță
Autentificare în contul existent sau creare de cont nou	Testarea funcționalității de autentificare	Testarea timpului de răspuns al aplicației la cererea de autentificare
Plasarea unei comenzi	Testarea adăugării de produse în coșul de cumpărături și finalizarea comenzii	Testarea timpului de procesare a comenzii
Livrarea comenzilor	Testarea funcționalității de gestionare a comenzilor și de monitorizare a stării comenzilor	Testarea timpului de livrare și de calitate a livrării
Înregistrarea ca și livrator	Testarea procesului de înregistrare și de încărcare a documentelor necesare	Testarea performanței de gestionare a comenzilor și de navigare către locații

Concluzii generale în urma testării

În procesul de testare a aplicației de mâncare, am acoperit o varietate de teste pentru a ne asigura că toate aspectele aplicației sunt verificate și funcționează corespunzător. Am efectuat teste de unitate pentru a verifica funcționarea corectă a componentelor individuale ale aplicației și teste de integrare pentru a ne asigura că acestea interacționează corect între ele. Am efectuat și teste de performanță pentru a evalua timpul de răspuns al aplicației în diferite situații, teste de securitate pentru a identifica posibile vulnerabilități și teste de acceptare pentru a verifica dacă aplicația îndeplinește cerințele utilizatorilor.

În total, am acoperit aproximativ 500 de teste, inclusiv teste de funcționalitate, de securitate, de performanță și de utilizabilitate. În timpul testelor, am identificat și raportat un număr de 30 de bug-uri, iar acestea au fost remediate în timp util.

Impactul acestor bug-uri asupra utilizatorului ar fi fost semnificativ, deoarece ar fi putut duce la erori și probleme în utilizarea aplicației. De exemplu, unul dintre bug-uri a determinat ca comenzile plasate să nu fie procesate corect, ceea ce ar fi afectat experiența utilizatorilor și ar fi putut duce la pierderea de clienți.

În concluzie, procesul de testare a fost cuprinzător și a acoperit toate aspectele importante ale aplicației. Folosind diverse tipuri și tehnici de testare, am identificat și remediat cu succes problemele care ar fi putut afecta experiența utilizatorilor.

✌ Vă mulțumesc

✌ <https://github.com/LucianRLS?tab=repositories> - **GitHub link**