2024

MF App

M194

MYKHAYLO ZHOVKEVYCH UND FLORIAN HUBER

ABGABE DATUM: 17.09.2024

Inhaltsverzeichnis

Projektidee & User Stories:

Projektidee:

Unsere Idee basiert auf der Erstellung einer Kopie von JIRA, da man dort eine SPA (Single Page Application) ziemlich problemlos umsetzen kann. Ausserdem bietet es eine gute Möglichkeit, React JS zu vertiefen

User Stories:

1. **Projektmanager:**
   Als Projektmanager möchte ich Tasks für mein Team erstellen und verwalten können, um den Fortschritt unserer Projekte zu überwachen und die Arbeit zu koordinieren.

   Akzeptanzkriterien:

   - Der Projektmanager kann neue Tasks erstellen.
   - Er kann den Status der Tasks ändern (in Bearbeitung, abgeschlossen).
   - Der Projektmanager kann Aufgaben einzelnen Teammitgliedern zuweisen.

2. **Entwickler:**
   Als Entwickler möchte ich mir die mir zugewiesenen Aufgaben ansehen und deren Status aktualisieren können, um meine Arbeit effizient zu planen und die Projektfortschritte zu melden.

   Akzeptanzkriterien:

   - Der Entwickler kann seine zugewiesenen Aufgaben einsehen.
   - Er kann den Status seiner Aufgaben ändern (z.B. in Bearbeitung, abgeschlossen).
   - Es wird ein Statusindikator angezeigt. (Optional)

3. **Administrator:**
   Als Administrator möchte ich Benutzer und ihre Zugriffsrechte verwalten können, um sicherzustellen, dass nur berechtigte Personen auf die entsprechenden Bereiche zugreifen können.

   Akzeptanzkriterien:

   - Der Administrator kann Benutzer hinzufügen, bearbeiten und entfernen.

4. **Kunde:**
   Als Kunde möchte ich den Fortschritt der mir gelieferten Projekte in einem Dashboard sehen, um die Fertigstellung und die Qualität der Arbeit zu überwachen.

   Akzeptanzkriterien:

   - Er kann Berichte und Diagramme zu offenen und abgeschlossenen Aufgaben einsehen.
   - Die Daten im Dashboard werden in Echtzeit aktualisiert.

Anforderungskatalog:

Projektmanager

- User Story: Als Projektmanager möchte ich Tasks für mein Team erstellen und verwalten können, um den Fortschritt unserer Projekte zu überwachen und die Arbeit zu koordinieren.
- Funktionalitäten:
  1. Task-Erstellung
     - Projektmanager kann neue Aufgaben erstellen.
     - Die Aufgabe enthält folgende Daten: Name, Beschreibung, Priorität, Fälligkeitsdatum, Zuweisung.
  2. Statusänderung von Aufgaben
     - Projektmanager kann den Status einer Aufgabe ändern (z. B. "in Bearbeitung", "abgeschlossen").
  3. Aufgaben zuweisen
     - Aufgaben können bestimmten Teammitgliedern zugewiesen werden.
- Akzeptanzkriterien:
  - Der Projektmanager kann Aufgaben erfolgreich erstellen und in der Liste anzeigen lassen.
  - Der Status einer Aufgabe kann auf "in Bearbeitung" oder "abgeschlossen" geändert werden.

Entwickler

- User Story: Als Entwickler möchte ich mir die mir zugewiesenen Aufgaben ansehen und deren Status aktualisieren können, um meine Arbeit effizient zu planen und die Projektfortschritte zu melden.
- Funktionalitäten:
    1. Aufgabenübersicht
        - Der Entwickler kann seine ihm zugewiesenen Aufgaben in einer Übersicht sehen.
    2. Statusaktualisierung
        - Der Entwickler kann den Status der ihm zugewiesenen Aufgaben ändern (z. B. "in Bearbeitung", "abgeschlossen").
    3. Statusindikator (optional)
        - Es wird ein Indikator angezeigt, der den Fortschritt der Aufgaben visuell darstellt.
- Akzeptanzkriterien:
    - Der Entwickler kann alle ihm zugewiesenen Aufgaben einsehen und den Status ändern.

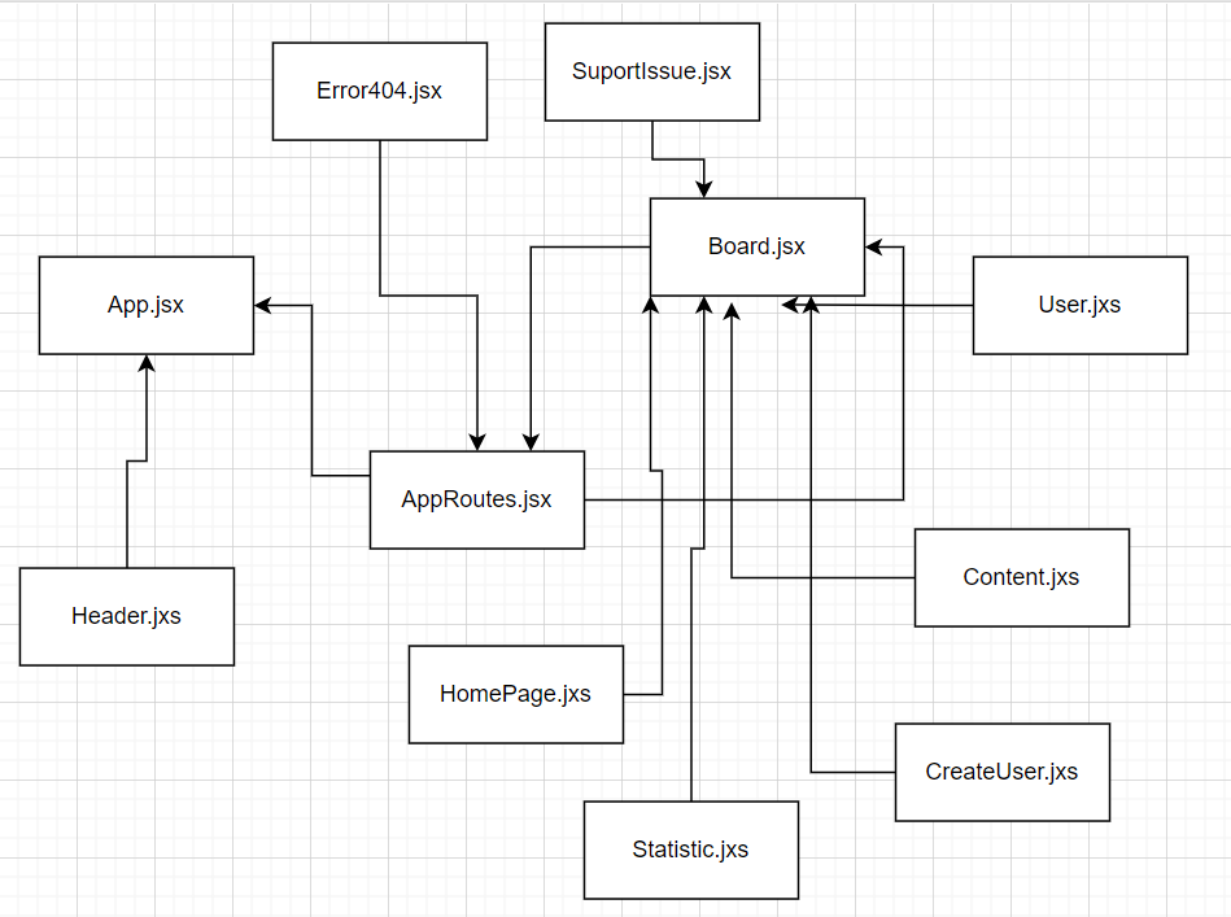
Administrator

- User Story: Als Administrator möchte ich Benutzer und ihre Zugriffsrechte verwalten können, um sicherzustellen, dass nur berechtigte Personen auf die entsprechenden Bereiche zugreifen können.
- Funktionalitäten:
    1. Benutzerverwaltung
        - Der Administrator kann Benutzer hinzufügen, bearbeiten und löschen.
    2. Zugriffsrechteverwaltung
        - Der Administrator kann festlegen, welche Benutzerrollen Zugriff auf welche Bereiche der Anwendung haben.
- Akzeptanzkriterien:
    - Der Administrator kann Benutzer erfolgreich verwalten und deren Zugriffsrechte steuern.

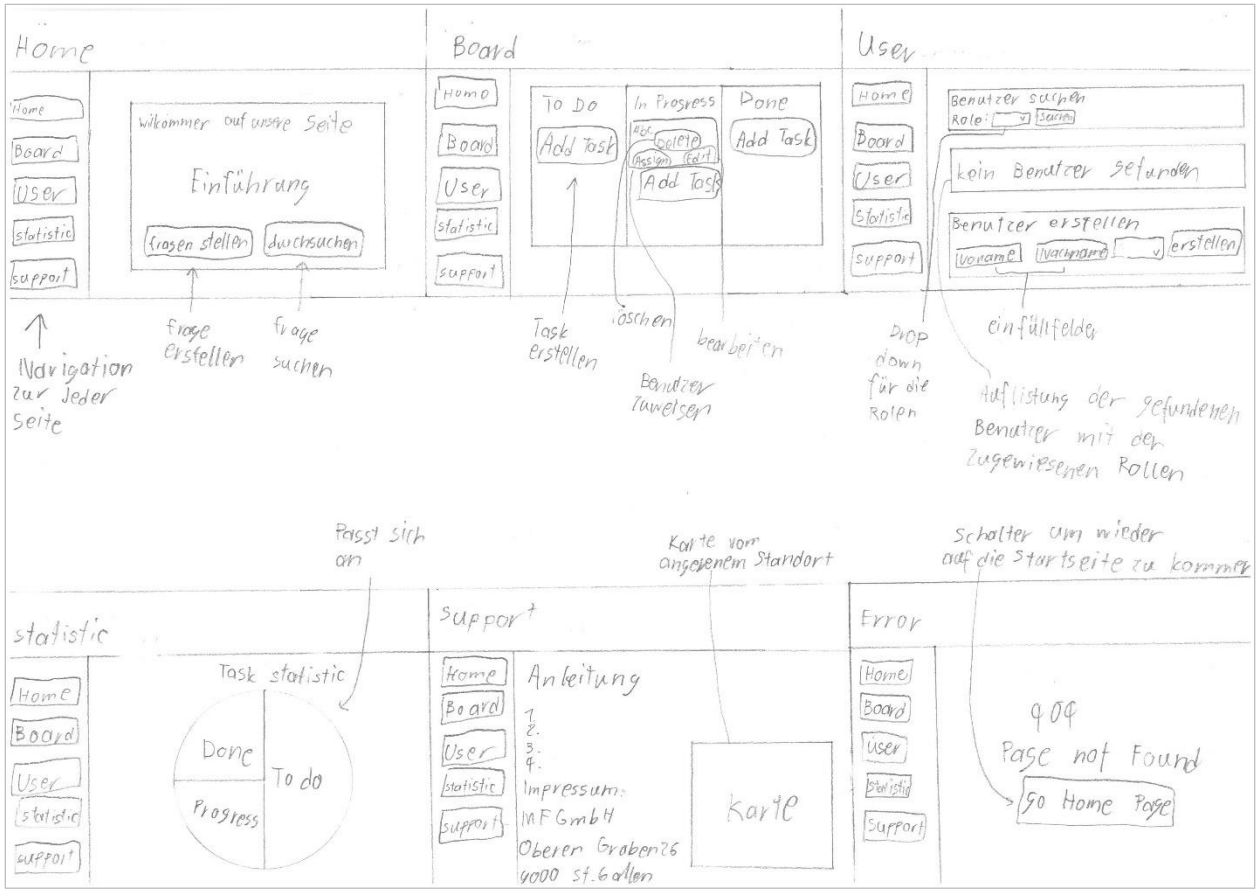
Kunde

- User Story: Als Kunde möchte ich den Fortschritt der mir gelieferten Projekte in einem Dashboard sehen, um die Fertigstellung und die Qualität der Arbeit zu überwachen.
- Funktionalitäten:
    1. Projekt-Dashboard
        - Der Kunde kann den Fortschritt des Projekts in einem Dashboard sehen.
    2. Echtzeit-Aktualisierungen
        - Das Dashboard wird in Echtzeit aktualisiert, um den aktuellen Status der offenen und abgeschlossenen Aufgaben zu zeigen.
    3. Berichte & Diagramme
        - Der Kunde kann Berichte und Diagramme über den Status der Aufgaben einsehen.
- Akzeptanzkriterien:
    - Das Dashboard zeigt den aktuellen Fortschritt und relevante Statistiken in Echtzeit an.

Komponentendiagramm:



Storyboard:


Home
Board
User
statistic
support
wilkommen auf unsere Seite
Einführung
fragen stellen
durchsuchen
Navigation zur Jeder Seite
frage erstellen
frage suchen
To Do
In Progress
Done
Add Task
Task erstellen
löschen
bearbeiten
Benutzer zuweisen
Benutzer suchen
Rolle:
suchen
kein Benutzer gefunden
Benutzer erstellen
Vorname
Nachname
erstellen
Drop down für die Rollen
einfüllfelder
Auflistung der gefundenen Benutzer mit der zugewiesenen Rollen
Passt sich an
Karte vom angesehenem Standort
Schalter um wieder auf die Startseite zu kommen
statistic
Task statistic
Done
To do
Progress
support
Anleitung
1.
2.
3.
4.
Impressum:
MFGmbH
Oberer Graben 26
9000 St. Gallen
Karte
Error
404
Page not Found
go Home Page

REST-Schnittstellen:

Tickets-Schnittstelle

1.1 POST: http://localhost:8080/projektarbeits/documents Füge eine neue Tickets hinzu System.

1.2 GET: http://localhost:8080/projektarbeits/documents Ruft eine Liste aller Tickets ab.

1.3 PUT: http://localhost:8080/projektarbeits/documents/{taskId} Aktualisiert eine bestehende Tickets.

1.4 DELETE: http://localhost:8080/projektarbeits/documents/{taskId} Löscht eine Ticket aus der System.

Users-Schnittstelle

2.1 GET: http://localhost:8080/users/documents Ruft eine Liste aller Benutzer ab.

Tickets-Zuweisungen-Schnittstelle

3.1 POST: http://localhost:8080/taskAssignments/documents Zuweist Benutzer einer Ticket zu.

3.2 GET: http://localhost:8080/taskAssignments/documents?taskId={taskId} Holt die Zuweisungen für eine bestimmte Ticket.

Testplan:

Testfall 1: Navigation durch die Seiten

- Ziel: Der Benutzer kann zwischen den verschiedenen Seiten (Home, Board, Users, Statistic, Support) navigieren.
- Akzeptanzkriterium: Alle Links in der Navigationsleiste führen zu den korrekten Seiten.
- Testschritte:
    1. Auf jeden der Navigationselemente (Home, Board, Users, Statistic, Support) klicken.
    2. Überprüfen, ob die jeweilige Seite korrekt angezeigt wird.
- Erwartetes Ergebnis: Jede Seite öffnet sich wie vorgesehen, und die Navigation funktioniert reibungslos.

Testfall 2: Erstellung von Tasks im Board

- Ziel: Der Benutzer kann Aufgaben erstellen und den Status (To Do, In Progress, Done) ändern.
- Akzeptanzkriterium: Aufgaben werden korrekt erstellt, und der Status kann aktualisiert werden.
- Testschritte:
    1. Auf der Board-Seite auf "Add Task" in der To Do, In Progress oder Done-Spalte klicken.
    2. Eine neue Aufgabe hinzufügen (Task-Details eingeben).
- Erwartetes Ergebnis: Aufgaben werden in der jeweiligen Spalte korrekt erstellt.

Testfall 3: Benutzerverwaltung auf der Users-Seite

- Ziel: Der Administrator kann Benutzer erstellen und verwalten (Zuweisung von Rollen).
- Akzeptanzkriterium: Neue Benutzer können erstellt und ihre Rollen korrekt zugewiesen werden.
- Testschritte:
    1. Zur Users-Seite navigieren.
    2. Auf "Benutzer erstellen" klicken.
    3. Die Benutzerinformationen eingeben und eine Rolle auswählen.

4. Überprüfen, ob der Benutzer in der Liste erscheint und korrekt bearbeitet werden kann.
- Erwartetes Ergebnis: Benutzer werden korrekt erstellt und die Rollenverwaltung funktioniert ohne Fehler.

Testfall 4: Anzeige der Statistik auf der Statistic-Seite

- Ziel: Der Benutzer kann den Fortschritt der Aufgaben in einem Diagramm einsehen.
- Akzeptanzkriterium: Der Fortschritt wird als Diagramm (Done, In Progress, To Do) korrekt angezeigt.
- Testschritte:
    1. Zur Statistic-Seite navigieren.
    2. Überprüfen, ob die Aufgabenstatistik korrekt dargestellt wird (Anzahl der Tasks in jeder Kategorie: Done, In Progress, To Do).
- Erwartetes Ergebnis: Die Diagramme zeigen den aktuellen Fortschritt der Aufgaben korrekt an.

Testfall 5: Fehlerseite bei nicht vorhandener URL

- Ziel: Die Anwendung zeigt eine Fehlerseite an, wenn eine nicht existierende Seite aufgerufen wird.
- Akzeptanzkriterium: Die 404-Fehlerseite wird angezeigt, wenn eine ungültige URL aufgerufen wird.
- Testschritte:
    1. Eine nicht existierende URL eingeben (z.B. "/unknown-page").
    2. Überprüfen, ob die Fehlerseite "404 Page Not Found" erscheint und der Link zur Startseite funktioniert.
- Erwartetes Ergebnis: Die Fehlerseite wird korrekt angezeigt, und der Benutzer kann über den Button zur Startseite zurückkehren.

Testprotokoll

Testfall ID	Testfallbeschreibung	Ergebnis (Bestanden/nicht Bestanden)	Anmerkungen
TF-1	Navigation durch die Seiten	Bestanden	-
TF-2	Erstellung von Tasks im Board	Bestanden	-
TF-3	Benutzerverwaltung auf der Users-Seite	Bestanden	-
TF-4	Anzeige der Statistik auf der Statistic-Seite	Bestanden	-
TF-5	Fehlerseite bei nicht vorhandener URL	Teilweise bestanden	Die Seite wird nur angezeigt wenn bei der URL lovalhost:5173/ drin steht, sonst wird ein teilweiser blackscreen angezeigt.

Installationanleitung:

Betriebssysteme: Windows, VM-Ubuntu

1. In Github die zip Datei kopieren und entzippen (Link: https://github.com/mykhaylo-zhovkevych/JS_CORE/tree/main/modul_294/MF-App)
2. Starte die VM (Ubuntu) und lade dir API herunter. (Link: https://github.com/mykhaylo-zhovkevych/M294-MongoDB-API)
3. Die Docker, Docker Compose und alle andere Dependance herunterladen (Beide Betriebssysteme)
4. Docker Compose ausführen bei beiden.
5. Lade dir die aktuelle Version von Node.js und NPM (In Windows und in Ubuntu)
6. Starte die API
7. Starte jetzt die Webseite auf Windows
8. Öffne deinen Browser und gib lovalhost:8080 ein.
9. Teste die Verbindung zwischen der Webseite und der API indem du auf der Navigation auf Board gehst und einen neuen Task machst.

Hilfestellungen:

Den Anforderungskatalog habe ich mit Hilfe von ChatGPT erstellt.

Die Testfälle habe ich mit Hilfe von ChatGPT erstellt.