

Inhalt der Checkliste

Projektanfrage Join.....	2
Allgemein.....	2
User Experience.....	2
Technische Anforderungen.....	2
Design.....	2
Responsiveness.....	3
Technische Umsetzung.....	3
Formulare.....	3
JavaScript / Clean Code.....	3
Vermeide diese häufigen Fehler.....	4
Funktionalitäten - User Stories (WIP).....	4
2. Benutzeraccount & Administration.....	4
Benutzerregistrierung.....	4
User Story 2.....	5
User Story 3.....	5
User Story 4.....	5
3. Kanbanboard & Taskmanagement.....	5
User Story 1.....	5
User Story 2.....	6
User Story 3.....	6
User Story 4.....	6
User Story 5:.....	7
User Story 6:.....	8
User Story 7:.....	8
4. Verwaltung der Kontakte.....	9
User Story 1:.....	9
User Story 2.....	9
User Story 3.....	9
User Story 4:.....	9
User Story 5:.....	10
5. Sonstiges.....	10
User Story 1:.....	10
User Story 2:.....	10

Projektabgabe **Join**

Bitte erfülle alle Punkte auf dieser Liste, bevor du das Projekt einreichst. Diese **Definition of Done (DoD)** kannst du für alle deine Projekte verwenden.

1. Zu Berücksichtigen

Allgemein

- ☐ Alle User Stories und Akzeptanzkriterien sind erfüllt.
- ☐ Alle Features funktionieren fehlerfrei und wie erwartet.
- ☐ Vor Abgabe werden mindestens 5 realistische Tasks und 10 Kontakte hinzugefügt.
- ☐ Alle Funktionalitäten wurden vor Abgabe von den Gruppenmitgliedern manuell getestet mit den aktuellsten Versionen der Hauptbrowser (Chrome, Firefox, Safari, Edge).

User Experience

- ☐ User erhält intuitiv Feedback bei Interaktionen (hover, toast-messages etc.)
- ☐ Alle UI-Elemente (Farben, Abstände, Schatten) entsprechen dem Design-Prototypen in Figma.
- ☐ Transitions auf anklickbaren Elementen liegen zwischen 75ms und 125ms.
- ☐ Join funktioniert auf mobilen Geräten und unterstützt vertikale Anordnung der Kanban-Spalten.
- ☐ Buttons haben die CSS Eigenschaft `cursor: pointer;`
- ☐ Inputs und Buttons haben keinen Standard-Border (Besser `border: unset;`);

Technische Anforderungen

- ☐ Join hat eine MPA-Architektur (Multi-Page-Application)
- ☐ Strukturierte und konsistente Dateinamen und -strukturen.
- ☐ Es gibt keine Konsolenfehler, Fehlermeldungen oder logs in der Konsole
- ☐ Maximal 400 Zeilen Code pro Datei.
- ☐ Erstellter Content ist unmittelbar sichtbar.

Design

- ☐ Haben Buttons die CSS Eigenschaft `cursor: pointer; ?`
- ☐ Inputs und Buttons haben keinen Standard-Border (Besser `border: unset;`);
- ☐ Form-Validation: Was passiert bei leeren Inputs?

Responsiveness

- ☐ Jede Seite funktioniert bei jeder Auflösung bis min. 320px (Fenster kleiner ziehen)
- ☐ Content-Begrenzung für große Monitore (max-width z.B. bei 1920px / linksbündig)
- ☐ Jede Seite funktioniert sowohl mobile als auch auf Desktop
- ☐ Standardmäßig ist der Landscape-Modus auf mobilen Geräten zu deaktivieren, außer er ist speziell optimiert.
- ☐ Keine horizontalen Scrollbalken bei kleineren Auflösungen

Technische Umsetzung

- ☐ Join mit mehreren separaten Seiten umsetzen (SPA vs. MPA) Multi-Page Application
- ☐ Dateinamen
 - ☐ beschreibend / aussagekräftig
 - ☐ konsistent
- ☐ Javascript Dateienstruktur / Projektstruktur
 - ☐ Für jede Seite mindestens eine JS-Datei
 - ☐ Eine allgemeine seitenübergreifende JS-Datei
- ☐ CSS- Dateienstruktur

Formulare

- ☐ Wurde eine Form Validation verwendet?
 - ☐ Erstellter Content sollte direkt zu sehen sein
 - ☐ Button deaktivieren während der Ladezeit
 - ☐ Assigned-to Feld
 - ☐ Das Drop-Down Menü muss sich wieder automatisch schließen, wenn neben das Drop-Down Menü geklickt wird
- Hinweis:** In diesem Feld sollten **Kontakte** ausgewählt werden können. Somit können potentielle Arbeitgeber deine Software deutlich besser testen, als wenn hier User ausgewählt werden.
- ☐ Subtask-Feld
 - ☐ Wenn **innerhalb** von dem Subtask-Feld auf Enter gedrückt wird, muss ein Subtask angelegt werden. Der Haupt-Task darf **nicht** erstellt werden.

JavaScript / Clean Code

- ☐ Eine Funktion hat nur eine Aufgabe
- ☐ Eine Funktion ist maximal 14 Zeilen lang (HTML ausgenommen)
- ☐ Deutliche Funktionsnamen
- ☐ Geschrieben in camelCase (Richtig: shoppingCart, falsch; Shopping_Cart) für Dateinamen, Variablen und Funktionen
- ☐ Der erste Buchstabe von Funktionen / Variablen ist **klein geschrieben**

- ☐ 2 Leerzeilen Abstand zwischen Funktionen
- ☐ Max 400 LOCs (Lines of Code) pro Datei
- ☐ Dateien sind richtig benannt: index.html, script.js, style.css
- ☐ Ggf. HTML Code in extra Funktion
- ☐ Extra Ordner für templates und Bilder (img)
- ☐ Statischer HTML Code wird **nicht** über JavaScript generiert
- ☐ Funktionen sind nach JSDoc Standard dokumentiert:
<https://jsdoc.app/about-getting-started.html>

Vermeide diese häufigen Fehler

- ☐ Menüpunkte verschieben sich nicht, wenn man drüber hovers
- ☐ Typische Fehler
 - ☐ Tickets verschwinden, wenn ich sie weiter ziehe
 - ☐ Kein User-Feedback, wenn etwas gespeichert / geändert wird
 - ☐ Columns in der Board-Übersicht gehen zu weit runter
 - ☐ Formvalidation bei Add Contact / Edit Contact fehlt
 - ☐ Kein "rauslaufen" von Subtasks, Kontakten und allgemeiner Content

Funktionalitäten - User Stories (WIP)

Es ist soweit, "Join" wartet darauf, von euch zum Leben erweckt zu werden! Bei diesem Kanban-Board-Projekt stehen uns zwei spannende Richtungen offen:

- Ein zentrales Board, auf dem alle Join-Mitglieder auf einem Board zusammenarbeiten können.

2. Benutzeraccount & Administration

Benutzerregistrierung

Als neuer Benutzer möchte ich mich registrieren können, um Zugang zu Join zu erhalten und Join nutzen zu können.

- ☐ Es gibt ein Registrierungsformular, auf dem Benutzer ihre E-Mail-Adresse, ihren Namen und ihr Passwort eingeben können
- ☐ Bevor die Registrierung abgeschlossen wird, muss der Benutzer die Datenschutzerklärung akzeptieren.
- ☐ Bei falscher Eingabe (z.B. ungültige E-Mail) erhält der Benutzer eine Fehlermeldung.
- ☐ Der "Registrieren"-Button ist deaktiviert, solange nicht alle Pflichtfelder ausgefüllt sind.

User Story 2

Als Benutzer möchte ich mich anmelden können, um Zugriff auf mein Dashboard und mein Kanban-Board zu bekommen.

- ☐ Es gibt ein Login-Formular mit Feldern für E-Mail und Passwort.
- ☐ Eine "Remember Me"-Checkbox ist vorhanden.
- ☐ Bei falscher Eingabe (z.B. falsches Passwort) erhält der Benutzer eine Fehlermeldung.
- ☐ Es gibt eine Option für einen Gast-Login (Dieser kann alle Funktionalitäten testen, z. B. das Bearbeiten/Löschen von Kontakten und Tasks).
- ☐ Nicht angemeldete Besucher von Join werden bei geschützten Seiten (Summary, Add-Task, Board, Contacts etc.) auf die Login-Seite weitergeleitet

User Story 3

Als Benutzer möchte ich mich von Join abmelden können, damit niemand ohne meine Zustimmung auf meinen Account zugreifen kann.

- ☐ Es gibt eine "Logout" -Option in der Benutzeroberfläche.
- ☐ Nach Auswahl dieser Option werde ich sicher aus der Anwendung ausgeloggt und zum Login-Bildschirm von Join weitergeleitet.
- ☐ Nach dem Abmelden sind meine persönlichen Daten und Einstellungen ohne erneutes Einloggen nicht zugänglich.

User Story 4

Als Benutzer möchte ich die wichtigsten Informationen zu Anzahl der Tasks in dem jeweiligen Status und den Task mit der nächsten Deadline auf meinem Dashboard sehen, wenn ich mich anmelde.

- ☐ Das Dashboard zeigt die Anzahl der Tasks bis zur nächsten Deadline an.
- ☐ Das Dashboard zeigt die Anzahl der Tasks in den Phasen ToDo, In Progress, Awaiting Feedback und Done.
- ☐ Abhängig von der Tageszeit wird eine Begrüßungsnachricht (z.B. "Good morning, [Benutzername]") angezeigt.

3. Kanbanboard & Taskmanagement

User Story 1

Als Benutzer möchte ich meine Tasks auf einem Kanban-Board angezeigt bekommen.



- ☐ Das Board hat ein Layout mit vier Spalten: ToDo, In Progress, Awaiting Feedback und Done.
- ☐ Wenn eine Spalte ohne Tasks steht hier eine Info, dass keine Tasks in dem jeweiligen Status sich befinden
- ☐ Jeder Task zeigt Kategorie, Titel, eine Vorschau der Beschreibung, alle zugewiesenen Benutzer mit Initialen und die Priorität des Tasks.
- ☐ Ich kann die vollständige Beschreibung und alle Infos zu einem Tasks anzeigen, wenn ich auf einen Task klicke.
- ☐ Es gibt ein "+"-Icon in jeder Spalte, dass das Hinzufügen eines neuen Tasks ermöglicht.

User Story 2

Als Benutzer möchte ich den Fortschritt von Tasks, die Subtasks enthalten, auf meinem Kanban-Board visualisiert sehen, um einen Überblick über den aktuellen Stand meiner Aufgaben zu haben.

- ☐ Jeder Task, der Subtasks enthält, zeigt eine Fortschrittsanzeige oder ein Balkendiagramm.
- ☐ Die Fortschrittsanzeige zeigt die Anzahl der erledigten Subtasks im Verhältnis zur Gesamtzahl der Subtasks.
- ☐ Bei vollständig abgeschlossenen Tasks mit allen erledigten Subtasks wird der Fortschrittsbalken als 100% voll oder in einer abhebenden Farbe dargestellt.
- ☐ Durch Hover oder Klick auf die Fortschrittsanzeige erhält der Benutzer eine detaillierte Übersicht, z.B. "5 von 7 Subtasks erledigt".

User Story 3

Als Benutzer möchte ich eine Suchfunktion nutzen können, um spezifische Tasks anhand ihres Titels auf meinem Kanban-Board schnell zu finden.

- ☐ Es gibt ein Suchfeld oder eine Suchleiste auf dem Kanban-Board.
- ☐ Bei Eingabe eines Suchbegriffs werden die Ergebnisse in Echtzeit gefiltert und angezeigt.
- ☐ Nur Tasks, deren Titel oder Beschreibung den eingegebenen Suchbegriff enthält, werden in den Suchergebnissen angezeigt.
- ☐ Bei einer leeren Suchanfrage oder beim Löschen des Suchbegriffs werden alle Tasks wieder angezeigt.
- ☐ Ein Hinweis oder eine Meldung wird angezeigt, wenn keine Tasks den Suchkriterien entsprechen, z.B. "Keine Ergebnisse gefunden".

User Story 4

Als Benutzer möchte ich Tasks auf verschiedene Weise, intuitiv hinzufügen können und dabei alle notwendigen Details angeben, um meine Arbeit effizient und organisiert auf dem Kanban-Board darzustellen.

- ☐ Es gibt ein "Add Task"-Option im Hauptmenü der Anwendung.
- ☐ Jede Spalte auf dem Kanban-Board hat ein "+"-Icon, durch das direkt ein neuer Task zur jeweiligen Spalte hinzugefügt werden kann. Der jeweilige Status der Spalte wird dem neuen Task direkt hinzugefügt
- ☐ Neben der Suchleiste befindet sich ein weiteres "Add Task"-Symbol, über das ein neues Task-Formular aufgerufen werden kann.
- ☐ Beim Klicken auf eines dieser Symbole oder Optionen öffnet sich ein Formular mit folgenden Eingabefeldern:
 - ☐ **Titel ***: Ein Pflichtfeld, in das der Name des Tasks eingegeben wird.
 - ☐ **Beschreibung**: Ein optionaler Input, um weitere Informationen zum Task zu geben.
 - ☐ **Fälligkeitsdatum (Due Date)***: um das geforderte Abschlussdatum des Tasks anzugeben.
 - ☐ **Priorität**: Eine Auswahl mit den Optionen "urgent", "medium" und "low". Defaultwert: "Medium" ist per Default automatisch vorselektiert.
 - ☐ **Zugewiesen an (Assigned to)**: Ein Dropdown-Menü um den verantwortlichen Benutzer oder das Teammitglied für den Task auszuwählen.
 - ☐ **Kategorie***: Ein Dropdown-Menü oder eine Suchleiste, um den Task einer bestimmten Kategorie zuzuweisen. Es gibt als Kategorien "Technical Tasks" und "User Story" zur Auswahl.
- ☐ Es muss mindestens der Titel, ein Fälligkeitsdatum und eine Kategorie definiert werden, um einen Task speichern zu können.

User Story 5:

Als Benutzer möchte ich Subtasks zu meinen Hauptaufgaben hinzufügen, bearbeiten und organisieren können, um meine Aufgaben detaillierter zu strukturieren und den Fortschritt besser nachverfolgen zu können.

- ☐ Im Task-Formular gibt es einen speziellen Abschnitt mit einem Eingabefeld für Subtasks.
- ☐ Bei Fokus auf dem Subtaskfeld wird durch Drücken der Eingabetaste oder durch Klicken das Häkchen-Symbol der eingegebene Subtask zur Liste der Subtasks für diese Hauptaufgabe hinzugefügt. Ein "X"-Symbol im Eingabefeld dient zum Zurücksetzen des Eingabefeldes, ohne einen Subtask hinzuzufügen.
- ☐ Nach dem Hinzufügen eines Subtasks wird das Eingabefeld automatisch geleert und steht für die Eingabe eines weiteren Subtasks bereit.

- ☐ Beim Überfahren (Hover) eines Subtasks mit der Maus werden ein Stift-Icon und ein "X"-Icon sichtbar.
 - ☐ Das Stift-Icon ermöglicht es den Benutzern, den Titel eines bestehenden Subtasks zu bearbeiten.
 - ☐ Das "Mülleimer"-Symbol ermöglicht es, einen bereits hinzugefügten Subtask zu löschen.

User Story 6:

Als Benutzer möchte ich die Möglichkeit haben, bestehende Tasks zu bearbeiten oder zu löschen, indem ich deren Detailansicht aufrufe, um Änderungen vorzunehmen oder nicht mehr benötigte Tasks zu entfernen.

- ☐ Ein Klick auf einen Task öffnet seine Ticketdetails-Ansicht.
- ☐ In der Ticketdetails-Ansicht gibt es eine Editieroption mit einem Stift-Icon, um den Bearbeitungsmodus zu aktivieren.
- ☐ Im Bearbeitungsmodus können alle Details des Tasks, wie z.B. Titel, Beschreibung, Fälligkeitsdatum, Priorität, Zugeordnete und Subtasks geändert werden. **ABER:** Kein Edit der Kategorie!
- ☐ Alle Änderungen, die im Bearbeitungsmodus vorgenommen werden, können gespeichert oder verworfen werden. Es gibt entsprechende Schaltflächen, um dies zu tun.
- ☐ In der Ticketdetails-Ansicht gibt es auch ein Papierkorb-Icon, um den Task dauerhaft zu entfernen. Bei Klick wird der Task gelöscht und nicht mehr auf dem Kanban-Board angezeigt.

User Story 7:

Als Benutzer möchte ich die Möglichkeit haben, Tasks per Drag & Drop zwischen den Spalten zu verschieben – sowohl auf Desktop- als auch auf Mobilgeräten – um den Status eines Tasks einfach und intuitiv zu aktualisieren.

- ☐ Jeder Task ist "greifbar", um ihn zwischen den Spalten zu verschieben.
- ☐ Während des Drag-Vorgangs wird eine visuelle Rückmeldung angezeigt, die dem Benutzer signalisiert, dass der Task bewegt wird. z.B leichte Drehung der Karte
- ☐ Das Loslassen des Tasks in einer Spalte platziert ihn in dieser Spalte und aktualisiert seinen Status entsprechend.
- ☐ Das Verschieben eines Tasks in eine neue Spalte sollte flüssig und ohne Verzögerung erfolgen.
- ☐ Nachdem ein Task in eine neue Spalte verschoben wurde, bleibt er an dieser Position, bis er erneut verschoben oder anderweitig aktualisiert wird.
- ☐ Jede Spalte auf dem Kanban-Board signalisiert visuell, dass sie einen Task aufnehmen kann, indem eine gestrichelte Box (dashed box) erscheint, wenn ein Task über sie gezogen wird (highlight).

- ☐ Mobil: Auf Mobilgeräten werden die Spalten vertikal angeordnet dargestellt. Benutzer können einen Task durch langes Drücken ("Long Tap") greifen und ihn dann nach oben oder unten ziehen, um ihn in eine andere Spalte zu verschieben oder **einfacher**: einen kleinen Pfeil in der oberen rechten Ecke öffnet ein Popup menü in welchem man auswählen kann wohin der Task verschoben werden soll.

4. Verwaltung der Kontakte

User Story 1:

Als Benutzer möchte ich eine übersichtliche Liste meiner Kontakte sehen, geordnet nach Buchstaben, um meine Kontakte leichter zu finden und ihre Details anzeigen zu lassen.

- ☐ Es gibt eine Seite oder einen Bereich für Kontakte
- ☐ Kontakte werden alphabetisch nach ihrem Namen sortiert und ihre E-Mail-Adresse unterhalb ihres Namens angezeigt.
- ☐ Die Liste ist in Abschnitte nach Buchstaben unterteilt, sodass Kontakte, die mit einem bestimmten Buchstaben beginnen, zusammen gruppiert sind.
- ☐ Ein Klick auf einen Kontakt öffnet eine Detailansicht mit dem Namen, der E-Mail-Adresse und der Telefonnummer des Kontakts.

User Story 2

Als Benutzer möchte ich Kontaktinformationen wie E-Mail und Telefonnummer nachschlagen können, um mich mit Kontakten in Verbindung zu setzen.

- ☐ Durch Klicken auf einen Kontakt in der Liste kann ich dessen Detailansicht aufrufen.
- ☐ In der Detailansicht werden alle gespeicherten Informationen wie Name, E-Mail-Adresse und Telefonnummer des Kontakts angezeigt

User Story 3

Als Benutzer möchte ich neue Kontakte zu meiner Liste hinzufügen können, um mit ihnen auf Join zu arbeiten

- ☐ Es gibt eine "Hinzufügen"-Option oder ein entsprechendes Symbol.
- ☐ Ein Formular öffnet sich, in dem ich Informationen wie Name, E-Mail und Telefonnummer eingeben kann.
- ☐ Nach dem Ausfüllen des Formulars und Bestätigen werden die Kontaktdaten gespeichert und in der Liste angezeigt.

User Story 4:

Als Benutzer möchte ich Kontakte bearbeiten oder löschen können, um meine Kontaktliste aktuell zu halten.



- ☐ In der Detailansicht jedes Kontakts gibt es Optionen zum Bearbeiten und Löschen.
- ☐ Die Bearbeiten-Option öffnet ein Formular mit den bereits gespeicherten Daten des Kontakts, welche angepasst werden können.
- ☐ Die Option 'Löschen' entfernt den Kontakt endgültig aus der Liste. Zudem wird der Kontakt aus allen Aufgaben entfernt, denen er zuvor zugewiesen war.

User Story 5:

Als Benutzer möchte ich auch meinen eigenen Account in der Kontaktliste bearbeiten können, um sicherzustellen, dass meine Daten aktuell sind.

- ☐ Auch mein eigener Account ist in der "Contacts"-Liste sichtbar.
- ☐ Ich kann auf meinen eigenen Kontakt klicken und ihn genauso bearbeiten wie andere Kontakte in der Liste.

5. Sonstiges

User Story 1:

Als Benutzer möchte ich die Rechtshinweise und Impressum von Join einsehen können, um Informationen über den Anbieter und den Nutzungsbedingungen zu erhalten.

- ☐ Es gibt einen Bereich wo man die Rechtliche Hinweise zu Join einsehen kann "Legal Notice"
- ☐ Durch Anklicken des Links werde ich zu einer Seite weitergeleitet, die alle notwendigen Informationen über den Anbieter und rechtliche Hinweise enthält.
- ☐ Verwendung von realitätsnahen Namen (kein Lorem Ipsum)

User Story 2:

Als Benutzer möchte ich die Datenschutzerklärung der Anwendung einsehen können, um zu verstehen, wie meine Daten verwendet und geschützt werden.

- ☐ Es gibt einen Bereich in Join, wo Benutzer die "Privacy Policy" einsehen können.
- ☐ Durch Anklicken des Links werde ich zu einer Seite weitergeleitet, die detaillierte Informationen darüber enthält, wie meine Daten gesammelt, verwendet und geschützt werden.