

PlantPal - Checkliste

Funktionalität

Bugs

- Wenn Bild nicht hochgeladen werden kann, dann ausgewähltes Bild löschen und default verwenden
- Nach Bestätigung der E-Mail direkt eingeloggt werden
 - redirect und set user
- Pflanze löschen -> confirm!
- Pflanze erstellen:
 - Fertig button mit cursor: pointer
- Pflanzenprofil:
 - Love (Herzen) in mobiler Ansicht anpassen
 - ml bei Wasserbalken mit anzeigen
 - Bild nicht croppen
 - Texte ändern:
 - "gieß mich mit" ändern in „pro Woche mit“
 - Emoji ändern so dass es in Firefox auch sichtbar ist
 - "Planze gießen" Button/Link nicht überall klickbar
- Dashboard:
 - Pal hinzufügen button entweder als float button oder im oberen Bereich als statischen Button

Bugs to fix

- Nach Bestätigung der E-Mail direkt eingeloggt werden
 - redirect und set user
- Pflanze löschen -> confirm!
- Pflanze erstellen:
 - Fertig button mit cursor: pointer
- Pflanzenprofil:
 - Love (Herzen) in mobiler Ansicht anpassen
 - ml bei Wasserbalken mit anzeigen
 - Texte ändern:
 - "gieß mich mit" ändern in „pro Woche mit“
 - "Planze gießen" Button/Link nicht überall klickbar

Fehlende Funktionen

- Allgemein:
 - Reminder
 - API
- Dashboard:
 - "Pal hinzufügen" button entweder als float button oder im oberen Bereich als statischen Button
 - mehrere Planzen gießen

- Pflanzenprofil:
 - Notizen zur Pflanze machen

Funktionen einbauen

- Allgemein:
 - Reminder
- Dashboard:
 - "Pal hinzufügen" button entweder als float button oder im oberen Bereich als statischen Button
- Pflanzenprofil:
 - Notizen zur Pflanze machen

Code Conventions

- rubocop zeigt einige linting-fehler an, welche wir fixen wollen
- rails_best_practices zeigt 15 Warnings an, welche wir fixen wollen

Security

- breakman hat 1 warning gefunden, welche wir beheben wollen

```
Confidence: Weak
Category: Unscoped Find
Check: UnscopedFind
Message: Unscoped call to `Plant#find`
Code: Plant.find(params[:id])
File: app/controllers/plants_controller.rb
Line: 90
```

Git

- ja, wir haben branches unseres Wissens sinnvoll verwendet
- gitflow besser, intensiver verwenden
- teilweise ja
- in Zukunft werden wir eine einheitliche Schreibweise mit Prefixes verwenden
- nein, keine API Key is verwendung

Deployment

- project ist online und [PlantPal](#) erreichbar
- wir können eine neue Version deployen

Test

- Eventuell für Login, wobei wir dieses mittels devise machen und die sich um Tests kümmern.
- keine Tests werden noch implementiert, weil wir nicht wissen wie. Sind aber offen für Verbesserungsvorschläge

DB + Models + Relationships

- ja, migrations wurden richtig verwendet
- ja, user und plants sind korrekt verknüpft
- man hätte alles mit scaffold generieren können
- ja, im plants_controller ist logik für das model

Routes + REST

- Ja, die Routes entsprechen den REST-Prinzipien
- Nein, wir haben PUT/PATCH verwendet um Daten zu verändern
- Unserer Ansicht nach, haben wir die üblichen Routes sinnvoll ergänzt

Controller

- ja, strong parameters werden richtig eingesetzt
- ja, before_action werden sinnvoll eingesetzt

Views

- Nein, laut rails_best_practices befindet sich kein Code im View, welcher ins Model/Controller gehört.