

WSF: Tripster

Allgemeine Informationen

Willkommen beim 'Tripster'-Projekt!

Diese Anwendung wurde entwickelt, um die Verwaltung von Gruppenausgaben zu vereinfachen und zu organisieren, insbesondere für Situationen, in denen mehrere Personen Kosten teilen – wie z. B. Reisen, Veranstaltungen oder gemeinsame Alltagsausgaben. Die App bietet eine saubere, intuitive Benutzeroberfläche, die es Benutzern ermöglicht, gemeinsame Ausgaben einfach zu verfolgen, zu verwalten und abzugleichen.

Wir freuen uns, dass Sie sich die Zeit genommen haben, dieses Projekt zu erkunden. Egal, ob Sie hier sind, um beizutragen, vom Code zu lernen, ein Problem zu melden oder einfach zu sehen, wie Dinge aufgebaut sind – vielen Dank! Ihr Interesse ist sehr wertvoll.

Das Projekt wird hauptsächlich mit JavaScript innerhalb eines Next.js-Frameworks entwickelt, zusammen mit Supabase für Authentifizierung und Datenbankdienste. Beiträge, Vorschläge und Feedback sind herzlich willkommen, und wir hoffen, dass Sie die App nützlich, aufschlussreich oder sogar inspirierend finden.

Projektstruktur

Das 'Tripster'-Projekt folgt einer modularen und skalierbaren Struktur, die für Klarheit und Wartbarkeit ausgelegt ist. Der Code ist hauptsächlich mit dem Next.js-Framework organisiert, mit logischer Trennung zwischen:

- pages/ — Handhabung von routenbasierten Komponenten
- components/ — Enthält wiederverwendbare UI-Elemente
- public/ — Statische Assets wie Logos und Bilder
- config/ — Datenbankkonfigurationsdateien

Wir verfolgen einen Mobile-First-Design-Ansatz mit responsiven Layouts, um ein reibungsloses Erlebnis auf allen Geräten zu gewährleisten.

Workflow im Projekt

Wir verwenden einen Feature Branch Git Workflow, der die Zusammenarbeit und Codequalität betont. Hier ist eine Zusammenfassung des Prozesses:

1. Jedes neue Feature, Bugfix oder jede Verbesserung wird in einem eigenen Branch vom develop-Branch entwickelt.
2. Entwickler reichen Merge Requests (MRs) ein, um ihre Feature-Branche in den develop-Branch zu mergen.
3. Nach erfolgreicher Überprüfung und Tests wird der develop-Branch in den main-Branch für Produktionsreleases gemerged.

Alle Beiträge durchlaufen einen Code-Review, um Qualität, Wartbarkeit und Einhaltung der Projektstandards zu gewährleisten.

Informationen zu Standard-Branches

Wir verwenden die folgenden Branches als Teil unserer Versionskontrollstrategie:

- main
 - Zweck: Produktionsbereiter Code
 - Berechtigungen: Nur Maintainer dürfen hier mergen
 - Bereitstellung: Wird für Live-Bereitstellungsbuilds verwendet
- develop
 - Zweck: Integration neuer Features und Staging-Umgebung
 - Berechtigungen: Alle Mitwirkenden können über geprüfte Merge Requests mergen
 - Bereitstellung: Wird für Tests und interne Vorschauen verwendet
- feature/*
 - Zweck: Branches für neue Features oder Aufgaben
 - Benennung: feature/ (z. B. feature/payment-page)
 - Berechtigungen: Offen für alle Mitwirkenden

Diese Struktur gewährleistet Stabilität im Produktionsbranch, während sie gleichzeitig aktive und flexible Entwicklung ermöglicht.

Beitrag

Der folgende Abschnitt bietet einen detaillierten Überblick über den Fortschritt und die Beiträge, die von jedem Projektmitglied während der Entwicklung der 'Tripster'-Anwendung geleistet wurden.

Contribution

The following section provides a detailed overview of the progress and contributions made by each project member throughout the development of the 'Tripster' application.

Scrum Master: Felix Bechtoldt

Implemented Features & Technical Contributions

- Created "Create Costs" Page
 - Integrated into the navigation bar
 - Applied design styling
- Linked All Application Pages
 - Ensured seamless navigation throughout the application
- Optimized "Create Costs" Page
 - Improved button functionality
 - Made the page fully responsive
- Supported Database Integration

- Provided assistance with database connectivity and setup
- Backend Development for "Edit Account" Page
 - Created profiles table
 - Linked profiles table with users and auth tables
 - Enabled editing of user data
 - Enforced verification for email address changes
 - Made the page responsive
- Handled Social Login Functionality
 - Social login accounts are detected correctly
 - Redirects users to the Google Account Management page
- Optimized "Edit Account" Page
 - Password confirmation required for changes
 - Current password required when updating email address
- Database Optimization
 - Updated usernames consistently across all related tables
- Supported Server Integration
 - Provided assistance with server connection and deployment
- Implemented "Show Password" Feature
 - Added to login, registration, and account edit pages

Scrum Activities

- Prepared, facilitated, and followed up on refinement sessions
- Monitored progress regularly
 - Continuously reviewed the team's work to ensure alignment with sprint goals and early detection of issues

Frontend Developer: Niklas Saaber

- Created and designed the "Friends" page (member overview)
- Designed and implemented user avatars
- Integrated avatars into the database and user profile pages
- Added the "Payments" page
- Applied a consistent design across the following pages:
 - Friends
 - EditAccount
 - InviteMembers
 - Name
 - Overview
 - Payments
 - Settings
 - Navbar
- Restored and redesigned the "Overview" page after faulty changes
- Fixed naming issues across the "Expenses", "Friends", and "Payments" pages
- Implemented logic behind the "Payments" page
- Ensured correct username passing across pages
- Enabled editing and deletion of expenses on the "Expenses" page
- Created the "DetailExpenses" page for detailed cost breakdowns

- Added a "Create Expense" button to the Navbar
- Disabled cost creation after group completion
- Created the "Friends" page to display and remove group members
- Reworked the "Overview" page:
 - Moved pie chart and add-expense button to the top
 - Added buttons for summary and expense list
 - Integrated an expenses list
 - Created the "Summary" page:
 - Linked from "Overview"
 - Shows how much the user owes or is owed
 - Optionally displays all expenses
 - Includes export function
- Implemented editing and deletion of individual expenses with permission restrictions
- Fixed expense calculation logic, including tracking total expenses per user
- Implemented "Expenses" feature to view all group expenses and edit own entries
- Enabled admins to edit and delete all expenses in a group
- Reworked pie chart for better visibility of user's share (visual and label adjustments)
- Implemented functionality to mark payments as completed (check-off system)
- Fixed navigation and label styling on the "DetailExpenses" page
- Fixed the add-expense button in the Navbar to correctly handle group_id
- Fully redesigned the layout of the "/addExpenses", "/editExpenses", and "/detailExpenses" pages for clean, mobile-first UI
- Implemented "Mark Group as Completed" feature to lock new expense creation and unlock payments
- Added payment summary showing what the user owes and is owed on the Payments page
- Implemented group reopening feature by admins to re-enable cost creation
- Removed overlapping numbers from the pie chart and improved visual clarity
- Added modern avatar options and a default avatar for users without selection

Backend Developer: Max Schmidt

- Created and designed the Edit Account page (frontend).
- Customized the button on the Edit Account page.
- Added a Navbar to the Edit Account page, including the Overview button.
- Built the Register and Login pages (functionality pending).
- Configured Supabase connection and integrated .env file.
- Enabled functionality for the Register page using the API-register route and Supabase.
- Implemented functional Login page with Supabase.
- Developed the Home page with session handling.
- Added a Logout button and session handling to the Edit Account page.
- Implemented a check on the Home page to see if the user is in the group, redirecting them to the Overview page if they are.
- Developed logic for the Name Page, generating a code and saving it to the database.
- Implemented logic for the Join Group page.
- Customized the Invite Member page with database connection and functionality.
- Developed the Add Expenses feature with database connection, full functionality, and design.
- Created the Expenses page.
- Developed the Settings page with full database connection and functionality.

- Added the tables.sql file containing the necessary tables.
- Fixed errors preventing the Navbar from appearing and made it possible to change the currency on the Add Expenses page.
- Restricted Settings page access to Admins only.
- Integrated Social Login for both Login and Register pages using Supabase, displaying usernames on the Edit Account and Expenses pages.
- Improved Session Handling on the Home Page.
- Added Session Handling to the Join Group page.
- Included Google Login logo on the Login and Register pages.
- Added password query and Register form fields.
- Integrated Session Handling on the Settings page.
- Replaced Alerts with Pop-Ups on Edit Account, Join Group, and Register pages.
- Fixed server bug causing issues with Social Login redirection.
- Added password validation for lowercase, uppercase letters, and numbers.
- Managed almost everything related to Supabase integration and configuration.
- Fixed Join Group pop-up error.
- Made the Delete Group button visible only to Admins and enabled account deletion functionality.
- Fixed Add Expenses button, Session Sharing, and restored the Group Delete button functionality.
- Handled Google Logout on the Home page via the Back Button.
- Added Back Buttons on the Join Group, Home, and Name pages.
- Changed Settings alerts to Pop-Ups.
- Adjusted Overview page (Compensation Button now pulls info from the Groups table), and redirected to Overview page after changing Vacation Mode in Settings.
- Completely set up the new server, integrated DNS, installed SSL certificate, and ensured the site runs and updates seamlessly.

Product Owner, Tester: Michael Schwarz

- loaded repository into GitLab
- created first basic pages (/name and /date) with javascript and css
- created page /overview (got later some major changes) with javascript and css
- created /joinGroup page (design got adjusted later) with javascript and css
- added and designed the navbar
- added react-icons for navbar and later for other pages
- made some overview fixes (added appRouter)
- created /settings (group settings) page with javascript and css
- added some package installation directions in the README
- made some design changes to the /start page (former /loginOderRegistrieren page)
- fixed a problem with the group settings button on the navbar
- added and designed pie chart (first without logic and real db data) using first with recharts, later than google charts
- removed misplaced json package files
- adjusted supabase connection in the providers.js to the newer supabase version
- reconstructed /addExpense with suspense loader (for build and hosting on the server)
- added logo for the app (shown at the tab as favicon)
- created and designed /friends (group member) page with javascript and css
- reconstructed /friends with suspense loader (for build and hosting on the server)

- added a placeholder for the pie chart, when no data is collected yet
- expanded the README with further instructions for the setup
- redesigned /editAccount page to match the rest of the app
- reconstructed /payments with suspense loader (for build and hosting on the server)
- reconstructed /editExpenses with suspense loader (for build and hosting on the server)
- reconstructed /detailsExpenses with suspense loader (for build and hosting on the server)
- fixed button in the /addExpense page
- fixed design of new add-expense button on the /overview page
- redesigned /editAccount (got changed later)
- redesigned whole on /payments page (twice)
- added a pop up window for leaving the group (not the last member) and changed the logic to differentiate between: just leaving (not last), leaving (last member) and deleting group
- renamed /loginOrderRegistriren page (our starting page) to /start
- adjusted order of inputs in /addExpense page
- adjusted order of inputs in /editExpense page

UI/UX, Frontend: Tarek Mazkatli

- Designed and developed the "Group Name" and "Invite Friends" pages from scratch.
- Redesigned and made the "Group Name" and "Join Group" pages responsive for optimal display across all devices.
- Visually restructured and redesigned the registration and login forms to create a cleaner and more intuitive user interface.
- Fine-tuned the responsive styling of the almost all pages, with a focus on enhancing usability on smaller mobile screens.
- Applied a completely new design to the Login and Register pages for a modernized look and feel.
- Developed a custom popup notification system to replace standard browser alerts, especially on actions like "add expense", for a more user-friendly experience.
- Implemented logic to detect registration attempts with already-used email addresses, triggering a relevant popup message.
- Replaced existing browser alerts on the "Edit Account" page with a centralized modal popup, providing consistent feedback for success, errors, and confirmation-required actions.
- On the Register page:
 - When invalid input is entered, a popup appears and disappears automatically after 5 seconds.
 - If the user corrects the input within those 5 seconds, the popup closes immediately.
- Dynamically displayed password strength and requirements as the user types a password during registration.
- Enhanced password management functionality by implementing ForgotPassword and UpdatePassword components:
 - Users can request password resets via email.
 - Secure, guided process for updating passwords.
- Improved the expense management module:
 - Integrated a user-friendly date picker for precise date selection when adding expenses.
 - Displayed the corresponding date next to each entry on the "Expense Content" page.
 - Allowed users to edit expense dates manually
 - Restricted full edit rights to admin only.

Backend / DevOps: Enzo Minkoley

- Initialized the project, structured the codebase, and set up the basic project architecture using Next.js.
- Created a Supabase instance, set up database tables, and configured the .env connection.
- Developed the home screen with session handling and redirection based on group membership.
- Revised the design and layout of all pages, optimized the UI, and implemented a consistent visual style.
- Identified and resolved numerous responsiveness issues across all device types.
- Replaced browser alerts on the login and registration pages with modern in-app notifications.
- Developed the overview page with dynamic display of all expenses retrieved from the database.
- Implemented the "Set Date" page with functional linking to the group session.
- Repaired the "Forgot Password" feature by integrating Supabase reset logic with the appropriate TH-Wildau domain and fixing email delivery via a custom SMTP server.
- Implemented dynamic password validation in the registration form, including visual indicators for password requirements.
- Fixed login issues related to group relationships and ensured proper session loading.
- Fully implemented the currency feature on the "Add Expenses" page with dynamic display.
- Enabled Row-Level Security (RLS) in Supabase and configured user-based access policies.
- Deployed and installed the project on the TH-Wildau server, ensuring stable live operation.
- Moderated most Sprint Reviews and actively led the meetings.
- Heavily involved in sprint planning, task distribution, and implementation throughout all project phases.

Frontend: Ilai Kolbe

- Created and designed the „Join group“ page
- Implemented addExpenses site with logic
 - be able to add expenses
 - show them in the overview page
- Afterwards fixed some interface issues with the expenses site
- implemented Logic in the friends page
 - see all members of the group with their name
 - see their avatars
 - one member has the admin status
 - as admin be able to kick other members of the group
- Fixed the friends button in the navbar to correctly transfer group_id
- Fixed pop-up error when navigating from addExpenses page to overview page
- Helped getting the Server connected
- Added the feature after you add an Expense you will be navigated to the overview page
- Helped adding the feature to join the group with a code
- Helped fixing the problem that the username you change in the profile page will be shown in the friends page
- Adjust the design on the name page
- Fixed problems with the navbar
 - adjust the navbar by different display sizes
 - changed the icons
- Reworked the payments page
 - be able to uncheck payments
 - be able to check payments back after you unchecked them

- implemented a button to undo the payment completed
- reworked the design of the page
- Added the expenses table to the database
 - with id, group_id, user_id, title, description, username, currency, label, cost, data and created_at

Final Words

The development of the cost-tracker 'Tripster' has been a collaborative effort driven by this motivated and diverse team. Throughout the project, we prioritized clarity, functionality, and user experience to create a tool that helps simplify shared expense tracking.

We would also like to acknowledge the support of ChatGPT, which was actively used during development to assist with structuring certain parts of the codebase and refining best practices. Additionally, this documentation was created with the help of ChatGPT to ensure clarity, consistency, and accessibility for future contributors and users.

Thank you to everyone who contributed their time, ideas, and code to this project. We're proud of the results and excited for what comes next—whether that's continued development, new features, or community involvement.

Issues and Contributors

Below is a comprehensive list of all issues and their respective contributors, as documented in GitLab:

ID	Title	Contributor
1	Page 'Homescreen' anlegen	Enzo Minkoley
2	Page 'Gruppenamen eingeben' anlegen	Tarek Mazkatli
3	Page 'Freunde einladen' anlegen	Tarek Mazkatli
4	Page 'Gruppenansicht' anlegen	Michael Schwarz
5	Page 'Gruppe beitreten' anlegen	Ilai Kolbe
6	Page 'Kosten erstellen' anlegen	Felix Bechtoldt
7	Page 'Datum einstellen' anlegen	Enzo Minkoley
8	Page 'Freunde' anlegen	Michael Schwarz
9	Page 'Freunde verwalten' anlegen	Niklas Saaber
10	Page 'Gruppe verwalten' anlegen	Michael Schwarz
11	Page 'Account verwalten' anlegen	Max Schmidt
12	Remove 'add friends' function	Tarek Mazkatli
13	Set up database for user management	Max Schmidt
14	Login screen	Max Schmidt
15	Sign up screen	Max Schmidt

ID	Title	Contributor
16	Create and handle group code	Max Schmidt
17	See and (manage) group members	Ilai Kolbe
18	Rewrite 'edit account' site	Felix Bechtoldt
19	Be able to add expenses	Ilai Kolbe
20	Responsive Design	Tarek Mazkatli
21	Overview rework	Niklas Saaber
22	Backend for object group	Max Schmidt
23	Site of all added costs	Ilai Kolbe
24	Site for summary	Niklas Saaber
25	Group listing site	Max Schmidt
26	(On Going) Dynamique UI/UX Design	Tarek Mazkatli
27	settingsSite	Max Schmidt
28	Host project on TH Wildau Server	Enzo Minkoley
29	Bug: Can't Join or Create Group	Michael Schwarz
30	See Password Requirements	Enzo Minkoley
31	(?) Enable RLS in supabase	Enzo Minkoley
32	Be Able to Leave Group	Michael Schwarz
33	Email Confirmation	Max Schmidt
34	Change Alerts to Pop-Ups	Tarek Mazkatli
35	Forgot Password	Tarek Mazkatli
36	Fix Missing Insert in Database	Max Schmidt
37	Create Labels	Tarek Mazkatli
38	Change Server Password	Michael Schwarz
39	Social Login/Register	Max Schmidt
42	Rewrite Code to Match New Database Structure	Max Schmidt
43	Edit Single Expenses	Niklas Saaber
44	Write project dokumentation	Michael Schwarz
45	Placeholder for Piechart	Michael Schwarz
46	Notifications Tab	Max Schmidt
48	Fix Expense Calculations	Niklas Saaber

ID	Title	Contributor
49	Delete Account	Max Schmidt
50	Date	Tarek Mazkatli
51	Open Expenses	Niklas Saaber
52	Currencies Need Functionality	Enzo Minkoley
53	Export Function	Felix Bechtoldt
54	Edit All Expenses as Admin	Niklas Saaber
55	Fix Forget Password	Enzo Minkoley
56	Overwork Piechart	Niklas Saaber
57	Mark payments as completed	Niklas Saaber
58	Fix detailsexpenses	Niklas Saaber
59	Fix addExpensesbutton	Niklas Saaber
60	Rework Expenses Design	Niklas Saaber
61	Show Sum of All Debts	Michael Schwarz
62	Group Settings Bug Fixes	Max Schmidt
63	mark group as completed	Niklas Saaber
64	'Back to Login/SignUp' Button	Max Schmidt
65	Pass over Admin Role	Max Schmidt
66	Reveal/Hide Password	Felix Bechtoldt
68	Documentation	Max Schmidt
69	Adjust Navbar	Ilai Kolbe
71	SERVER BUG: Can't Copy Code	Max Schmidt
72	Un-check Payments	Ilai Kolbe
73	BUG: Session Problem While Logged In	Max Schmidt
74	/addExpenses	Niklas Saaber
75	/editExpenses	Niklas Saaber
76	/detailExpenses	Niklas Saaber
77	Delete Group on Completion	Max Schmidt
78	Logging in issue with groups	Enzo Minkoley
79	summary of how much money you receive	Niklas Saaber
80	Re-open Group	Niklas Saaber

ID	Title	Contributor
81	BUG: Visible Problem Piechart	Niklas Saaber
82	Change Alert to Pop-Up Leaving Group	Michael Schwarz
84	Adjust Leave Group Pop-Up	Michael Schwarz
85	Get the server running	Max Schmidt
86	Adjust Payment Design	Michael Schwarz
88	Write CONTRIBUTING	Michael Schwarz
89	Milestone Plan	Michael Schwarz
90	Website	Michael Schwarz
91	Rework piechart	Niklas Saaber
92	Rework Avatar	Niklas Saaber
93	BUG: Join Group Does Not Load with 1st Time	Michael Schwarz
94	Rename Starting Page	Michael Schwarz
95	BUG: Group Status Inconsistent	Max Schmidt
96	Rework /registrieren	Tarek Mazkatli
97	More Piechart Functionality	Niklas Saaber
98	BUG: Old Avatars	Niklas Saaber
99	BUG: Wrong Colors for Button	Max Schmidt
101	Adjust Wrong-Input Error Messages	Tarek Mazkatli
102	5s Pop Ups (Wrong/Missing) Input	Tarek Mazkatli
103	Better Password Requirements	Tarek Mazkatli
104	BUG: New Users Getting Old Avatars	Niklas Saaber
105	Restore Database Supabase	Max Schmidt
106	Rework Whole Design	Niklas Saaber
107	Implement More Languages	Max Schmidt
108	Activity / Market Tab (Map)	Enzo Minkoley
110	New Logo	Michael Schwarz
111	Design Template	Tarek Mazkatli
113	Home Screen	Tarek Mazkatli
114	Login/Sign-Up	Tarek Mazkatli
115	Group Selection (Adjust later for group lists)	Max Schmidt

ID	Title	Contributor
116	Group Creation (Adjust later for group lists)	Max Schmidt
117	Expenses	Niklas Saaber
118	Navbar	Niklas Saaber
119	Expenses	Tarek Mazkatli
120	User Profile	Niklas Saaber
121	Group Settings	Ilai Kolbe
122	Invoice	Niklas Saaber
123	Maps Tab (for later implementation, maybe don't touch this yet)	Enzo Minkoley
124	Notifications	Max Schmidt
125	Members	Niklas Saaber
126	Uniform Password Requirement	Michael Schwarz
127	handleGoogleLogin/Register	Max Schmidt
128	Forgotten password page "Back to login button"	Ilai Kolbe
130	New Avatars	Niklas Saaber
131	Admin Center	Enzo Minkoley
132	BUG: Pop-Up Thrown Incorrectly	Max Schmidt
133	App Settings	Niklas Saaber
134	BUG: Change Avatar	Niklas Saaber
135	BUG: Fix more Language	Max Schmidt
136	BUG: Fix editAccount Cancel Button	Max Schmidt
137	Overview	Max Schmidt
138	test on server	Max Schmidt
139	BUG Fix Design on settings-side	Niklas Saaber
140	BUG: friends-side leave group button	Max Schmidt
141	BUG: Users can still create cost centers after ending their vacation	Max Schmidt
142	unify design on all sites	Ilai Kolbe
143	Terms and Conditions	Michael Schwarz
144	Design: Forgot Password / Update Password	Michael Schwarz
145	Loading site	Niklas Saaber
146	Adjust Start, Login and Register Page (Font, Color)	Michael Schwarz

ID	Title	Contributor
147	BUG: Wrong Pop-Up for Delete Expense	Tarek Mazkatli
148	Unify Pop Ups	Tarek Mazkatli
149	Funny: Pre-Defined Group Name	Ilai Kolbe
150	BUG: editExpenses	Tarek Mazkatli
151	BUG: Vacation Can Not Be Ended/Re-activated	Max Schmidt
152	Groups are searchable by name.in the sidebar	Max Schmidt
153	editLabelWindow-Design	Ilai Kolbe
154	Edit Text Color	Michael Schwarz
155	Edit Text Color (Expenses)	Michael Schwarz
156	Edit Log-Out Button	Michael Schwarz
157	detailedExpenses Site update Design	Ilai Kolbe
158	Edit Sidebar	Niklas Saaber
159	Create Labels 2.0	Tarek Mazkatli
160	App Wrapper	Max Schmidt
161	balance steht in der mitte und nicht linksbündig	Niklas Saaber
162	BUG: expensesContent Language	Max Schmidt
163	Align Overview Design & Update Translations in All Languages	Niklas Saaber
164	AppWrapper	Max Schmidt
165	Caching	Max Schmidt
166	Adjust Design AddExpense	Niklas Saaber
167	Notifications	Max Schmidt
168	Safe Right Language	Max Schmidt
169	dark/light Mode	Niklas Saaber
170	Old Sidebar Account Page	Niklas Saaber
171	Adjust Button Login	Michael Schwarz
172	BUG: Update Password Not Working	Max Schmidt
173	Update User Avatars to Match New App Design	Niklas Saaber
174	App Notification Double	Max Schmidt
175	Social login Bug	Max Schmidt
177	Responsive Design	Tarek Mazkatli

ID	Title	Contributor
178	BUG: Fix Fill Out Button	Ilai Kolbe
179	Fix Light Theme	Niklas Saaber
180	PayPal Link Payments	Niklas Saaber
181	BUG: Settings Vacation end error	Max Schmidt
182	BUG: No pushNotification wen vacation ended	Max Schmidt
183	BUG: Stuck in /home	Niklas Saaber
184	BUG : settings	Tarek Mazkatli
185	BUG: Navbar Slides Under Map	Niklas Saaber
186	BUG: Same Location Overlapping	Niklas Saaber
187	BUG: Can't Edit Expenses	Max Schmidt
188	BUG: Wrong Pop Up editAccount	Tarek Mazkatli
189	BUG: Stuck on Landing Page	Max Schmidt
190	BUG: Wrong Text Color Expenses	Tarek Mazkatli
191	BUG: Export Functions Don't Work	Felix Bechtoldt
192	BUG: Label is Missing in detailExpenses	Michael Schwarz
193	BUG: Payment Infos are Missing on detailFriends Page	Niklas Saaber
194	BUG: Payment Infos are Missing on detailFriends Page	Niklas Saaber
195	Small Fix: Name Too Long	Michael Schwarz
196	Cache Handler	Michael Schwarz
197	BUG: Edit Address	Niklas Saaber
198	BUG: Can't Edit Expense	Max Schmidt
199	BUG: No PopUP	Tarek Mazkatli
201	BUG: false logout on landing page	Max Schmidt
202	Missing Light Mode and Button Colors	Niklas Saaber