



Progettazione di app REACT

Presentazione progetto

Mathias Dalla Palma - VR504159



Obiettivi

- Dashboard digitale per la gestione dei piani di esercizi fisici.
 - - definire e personalizzare gli allenamenti e i singoli esercizi
 - organizzazione dei workout su calendario
 - inserimento e modifica centralizzata degli utenti
 - completo controllo su tutti i dati dell'applicazione

- UI intuitiva e pulita
- Responsive design

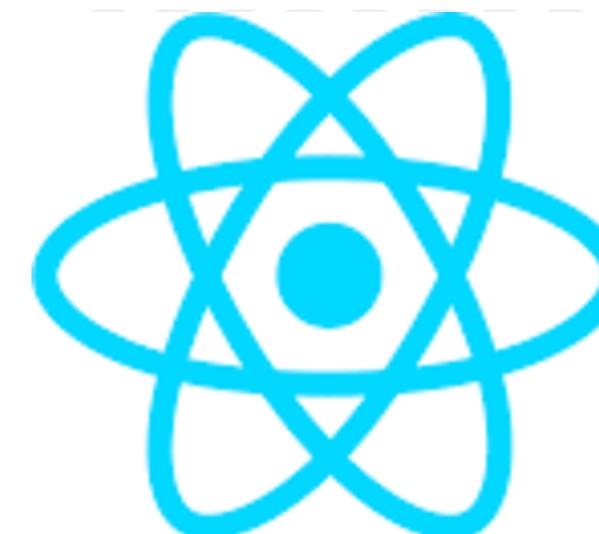
- Componenti react
- Routes

- Login
- Autorizzazioni

Strumenti utilizzati



FastAPI



React

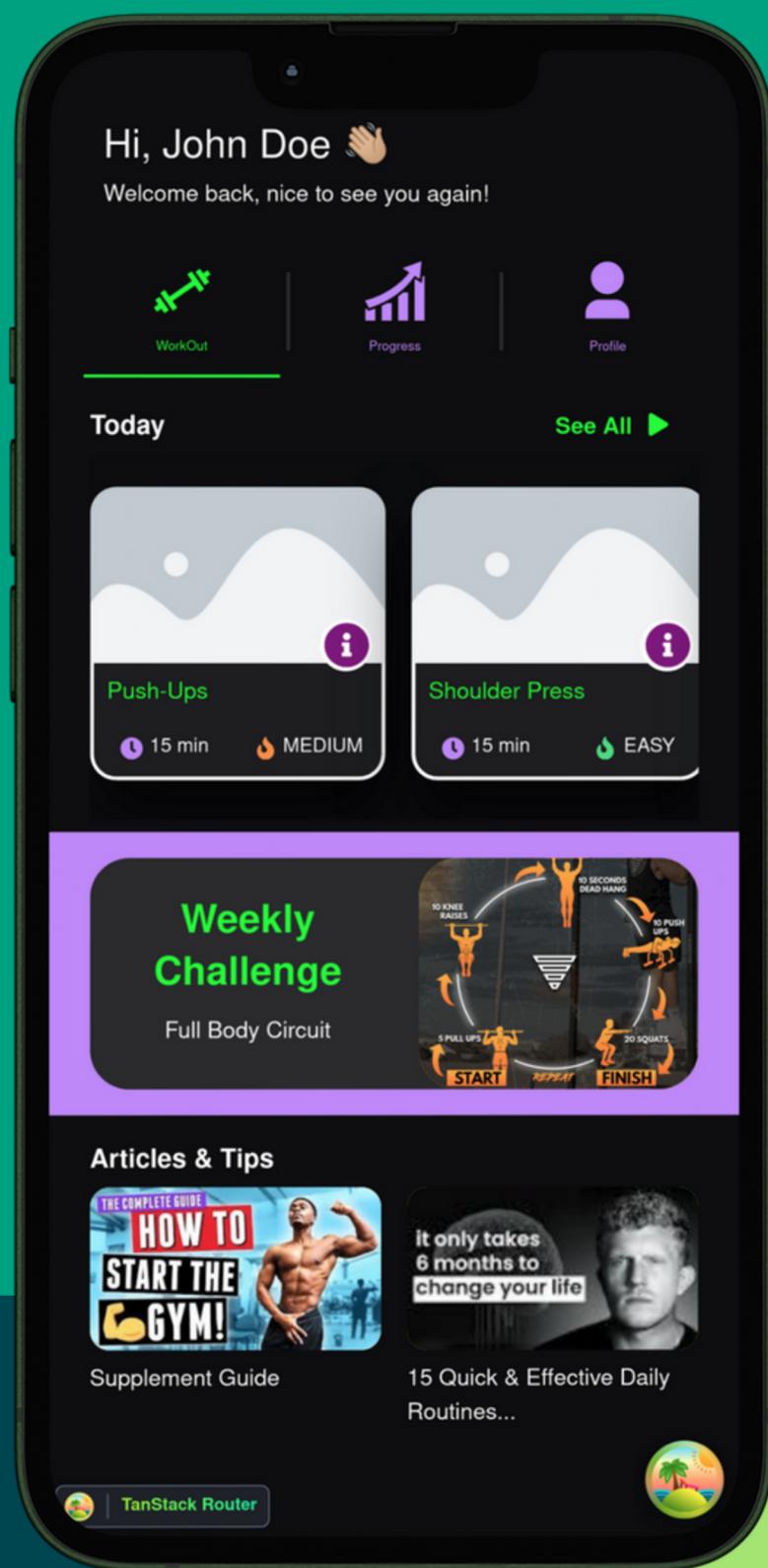


Cloud
Firestore

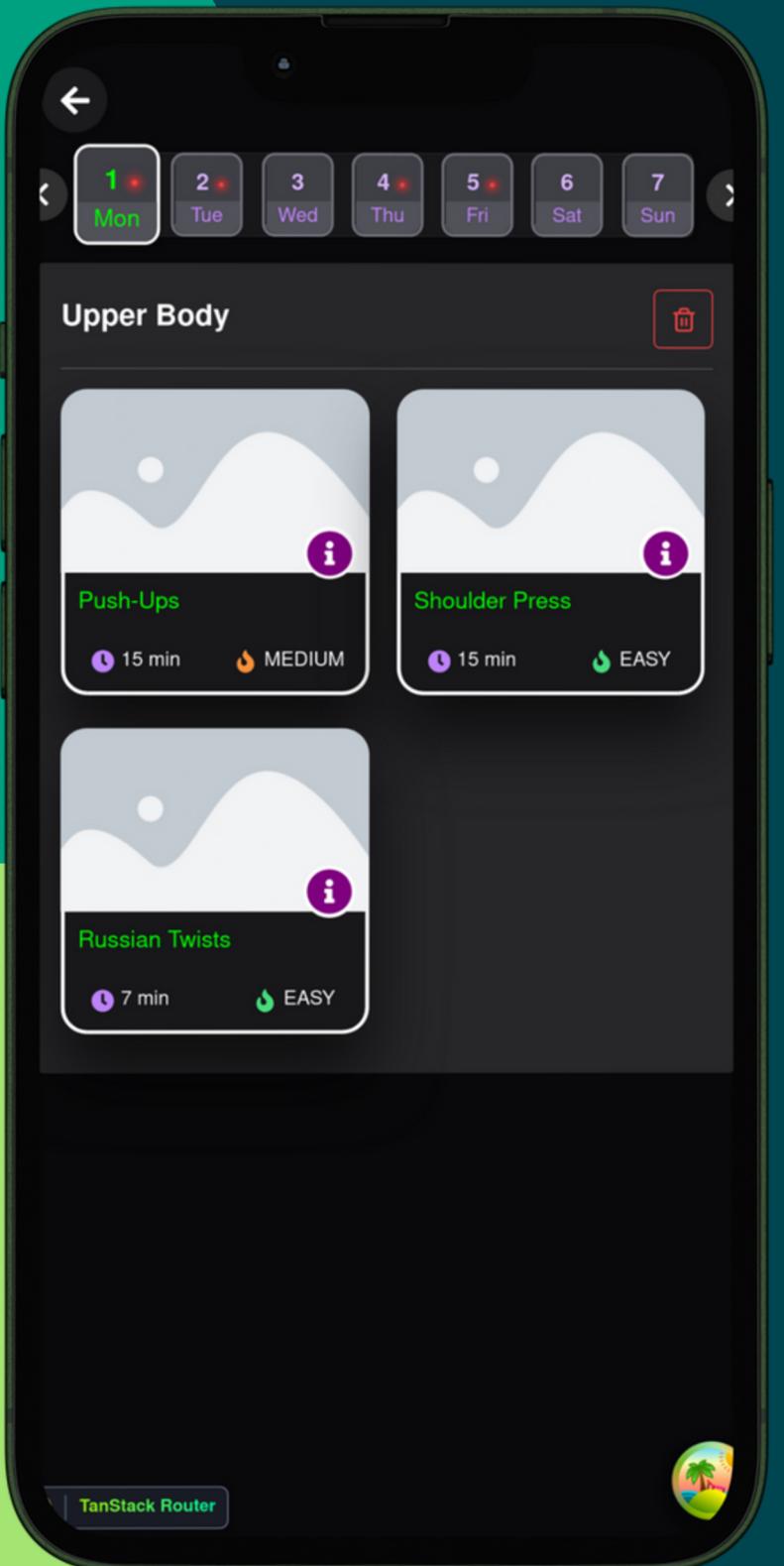
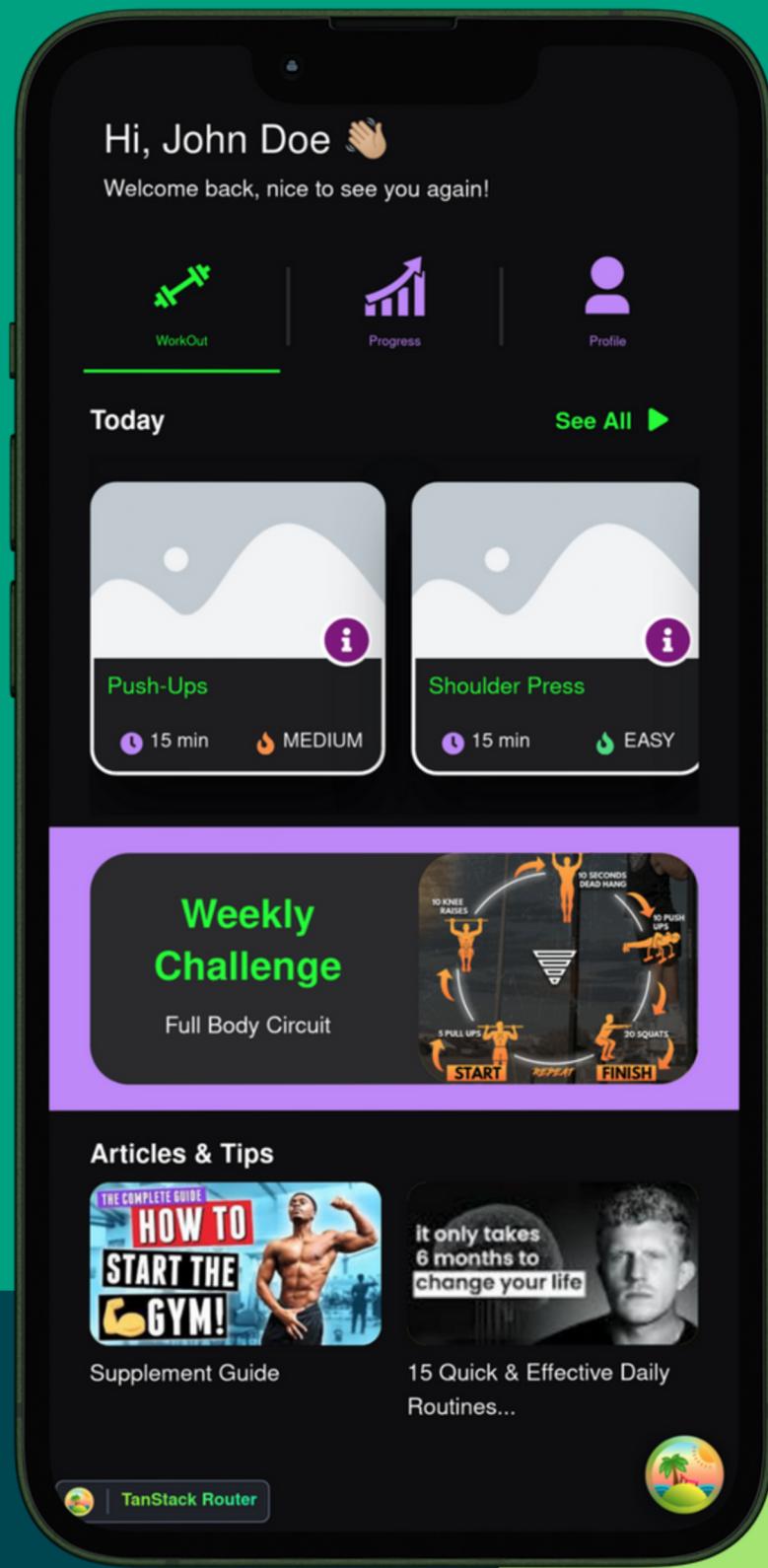


chakra

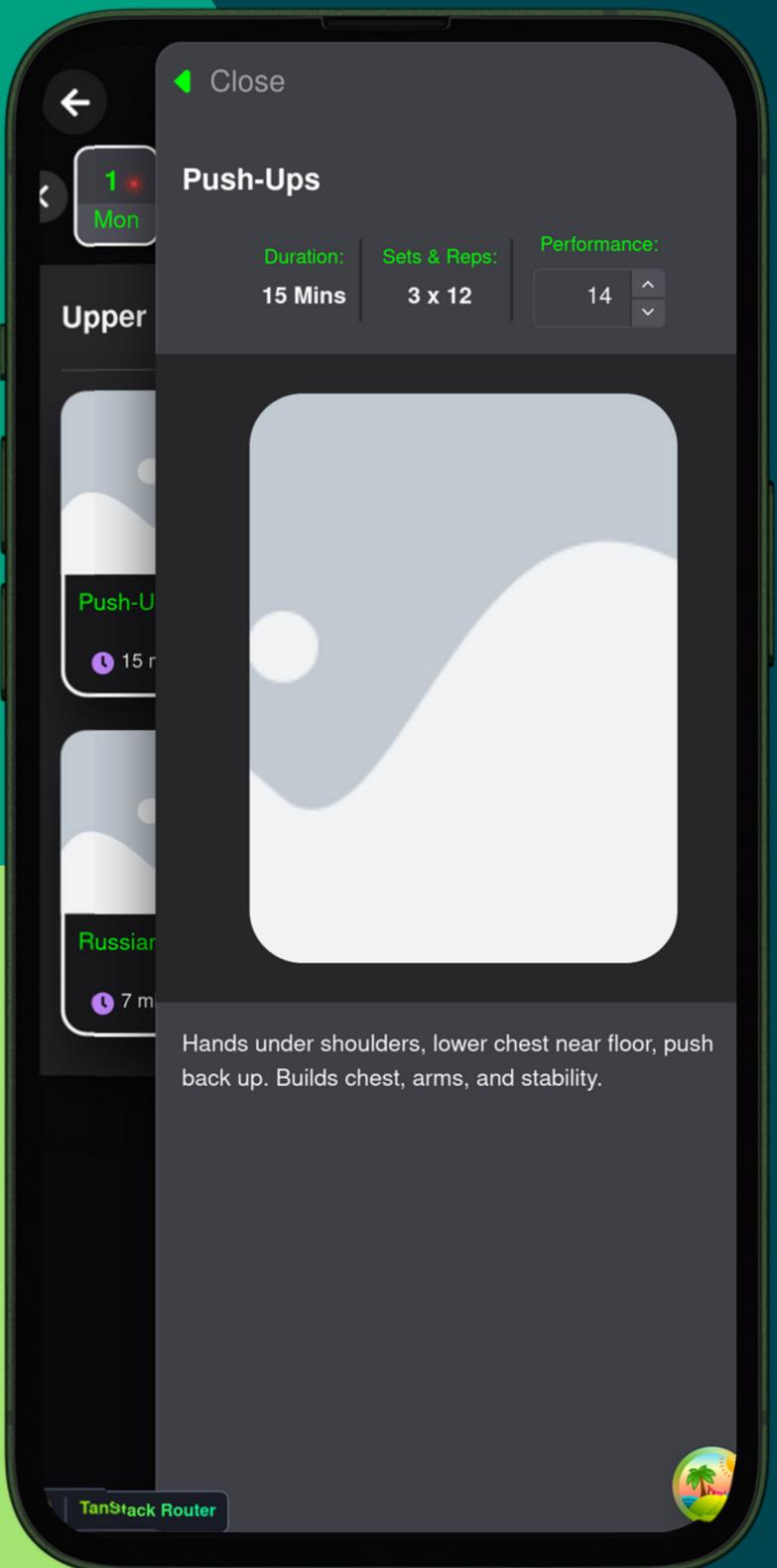
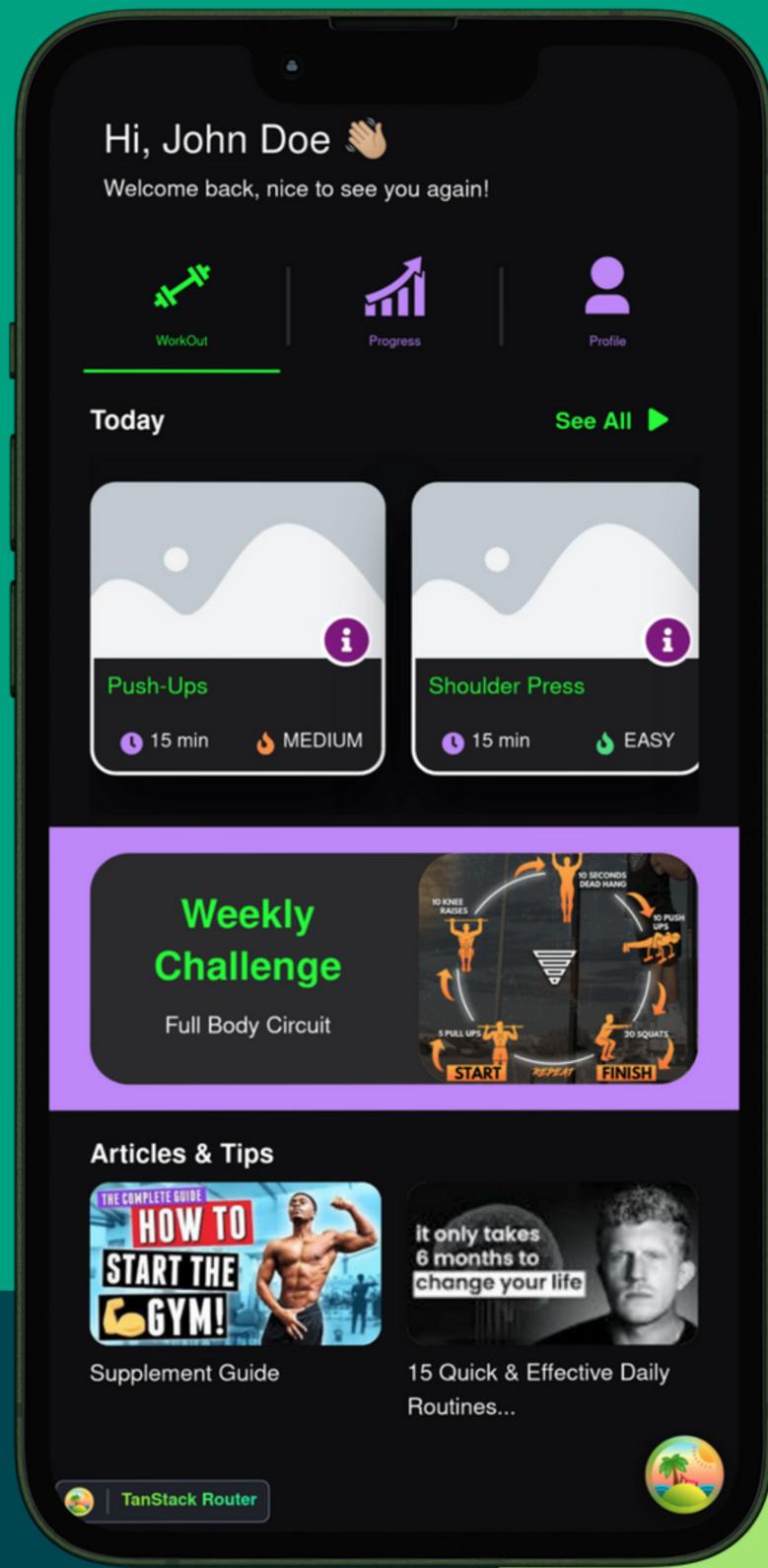
Design



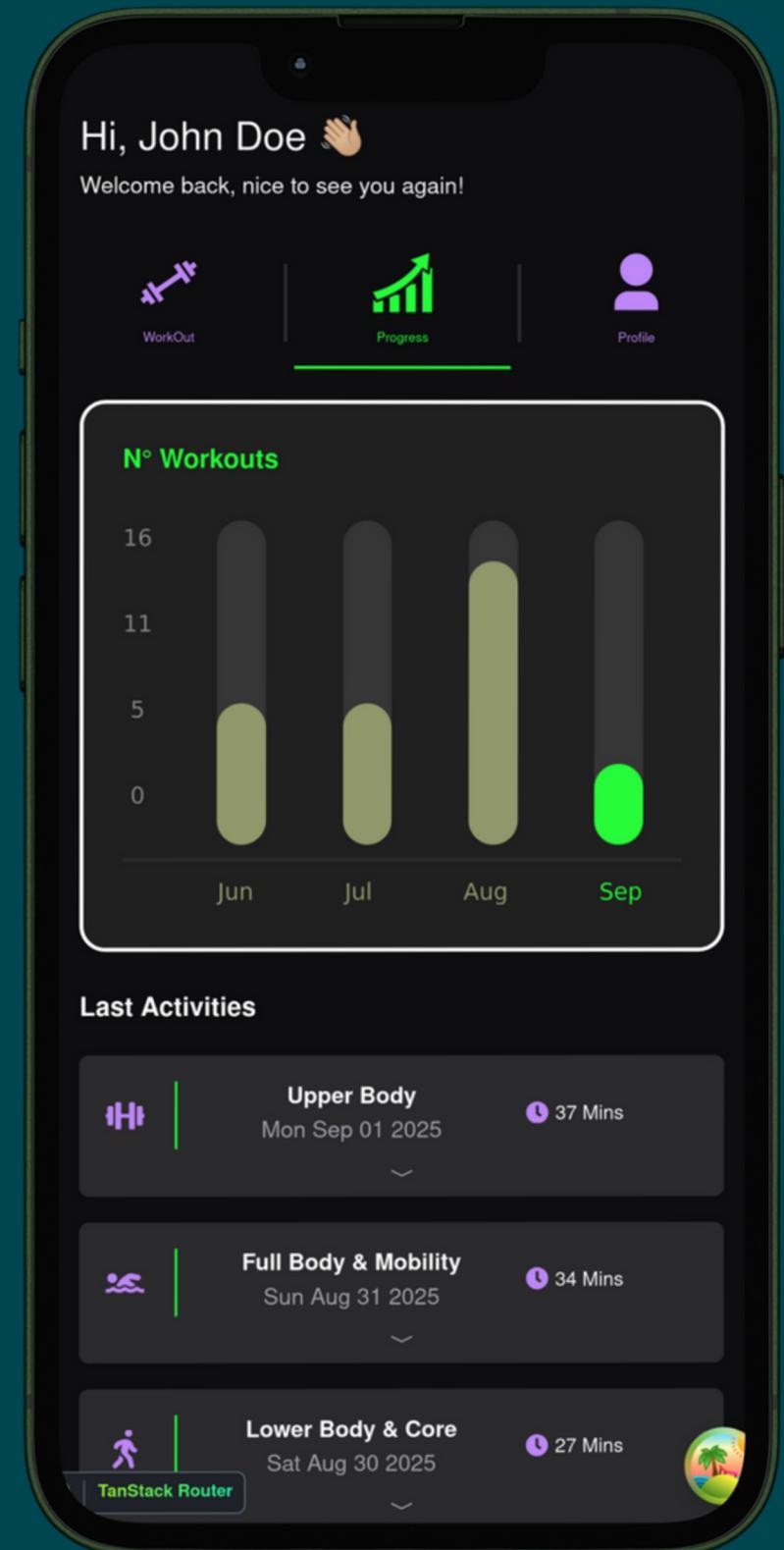
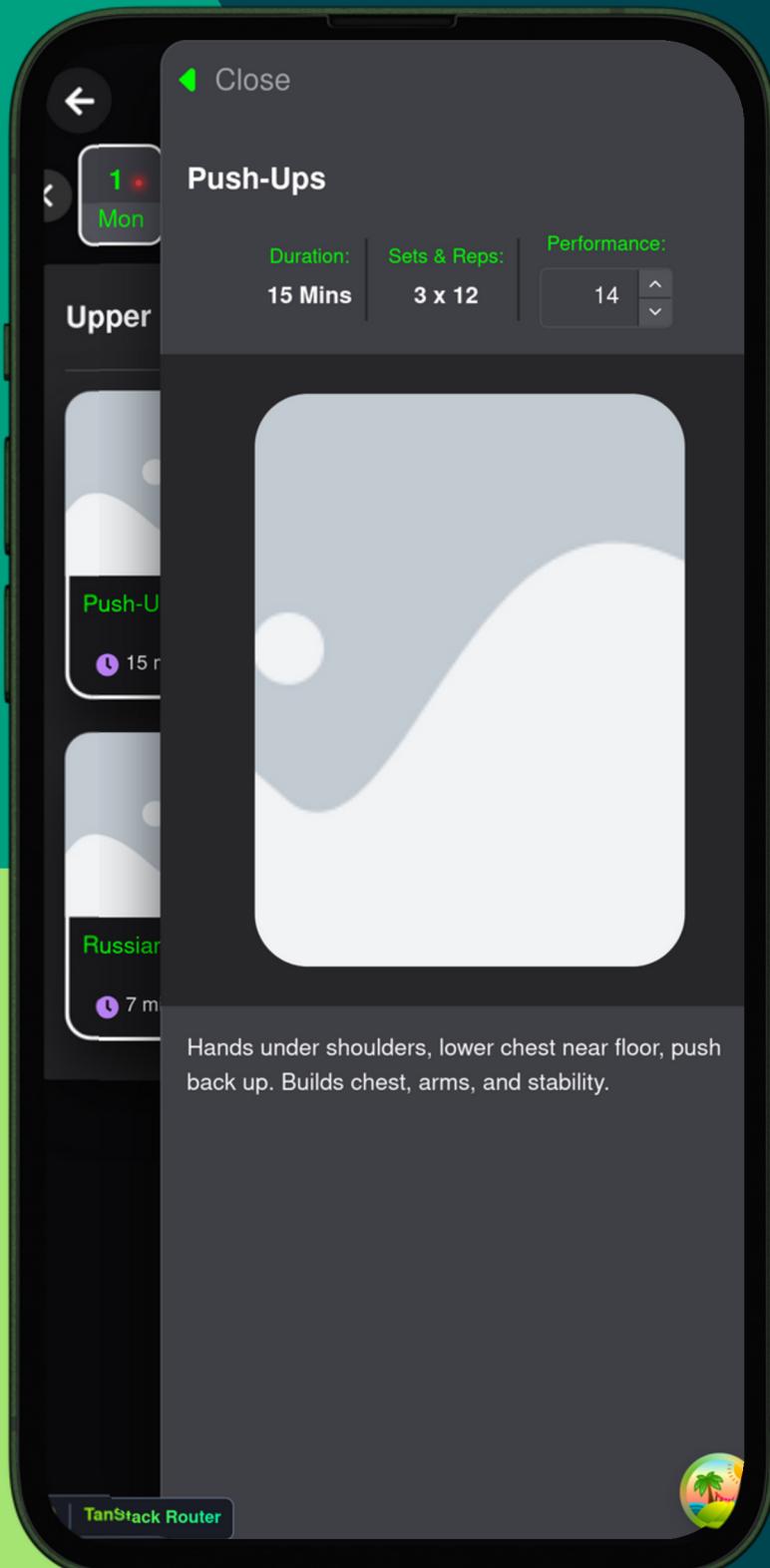
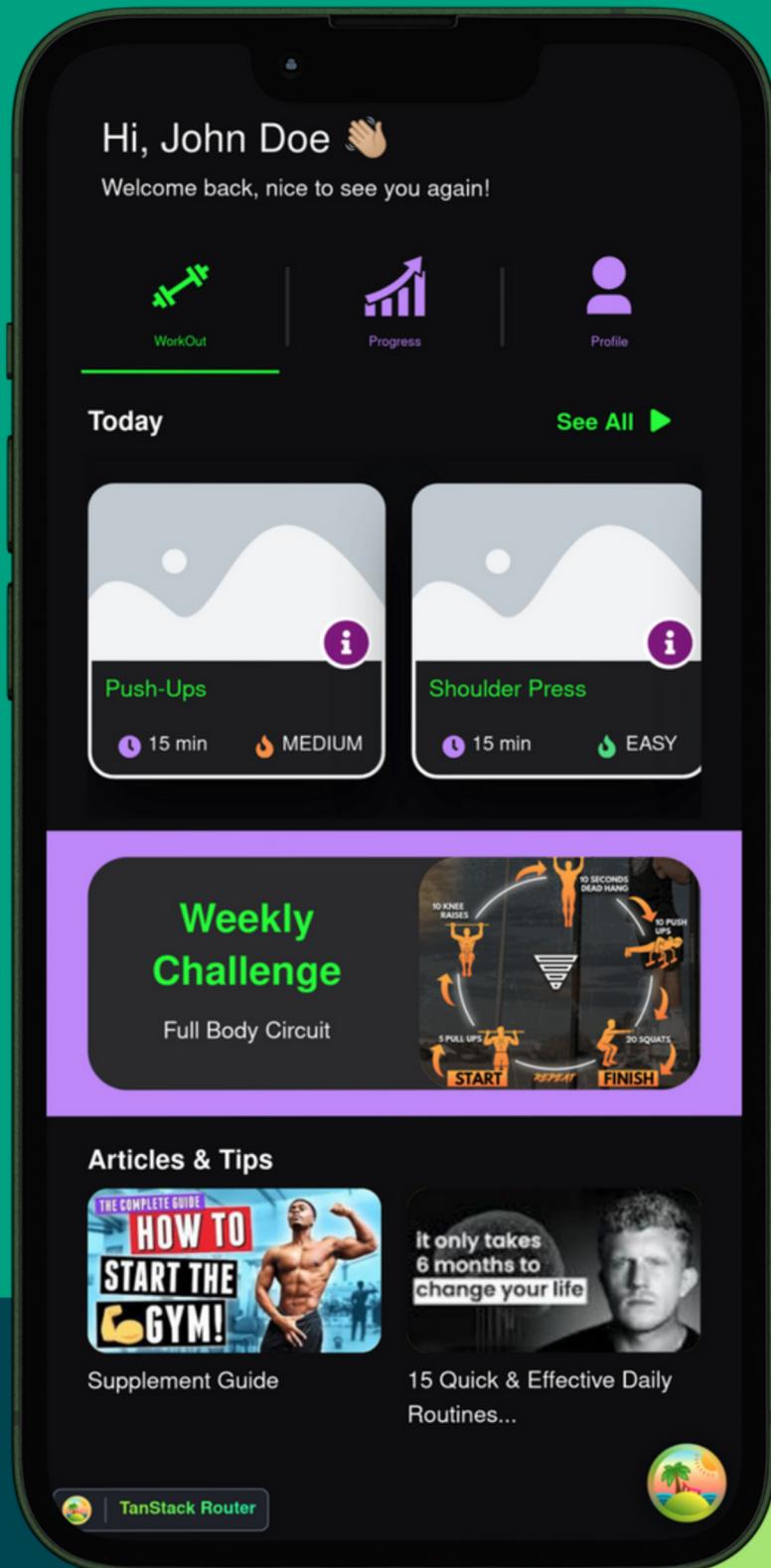
Design



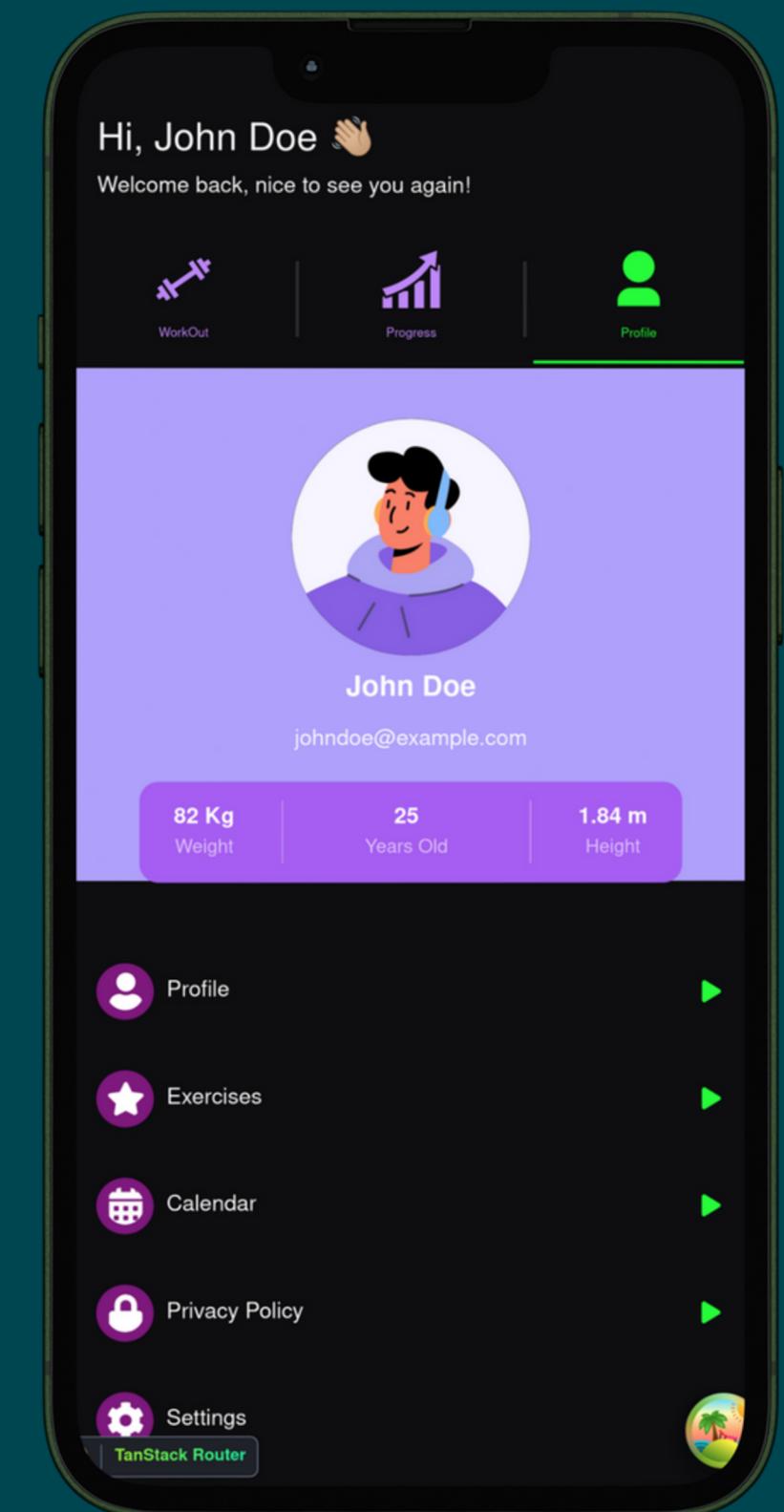
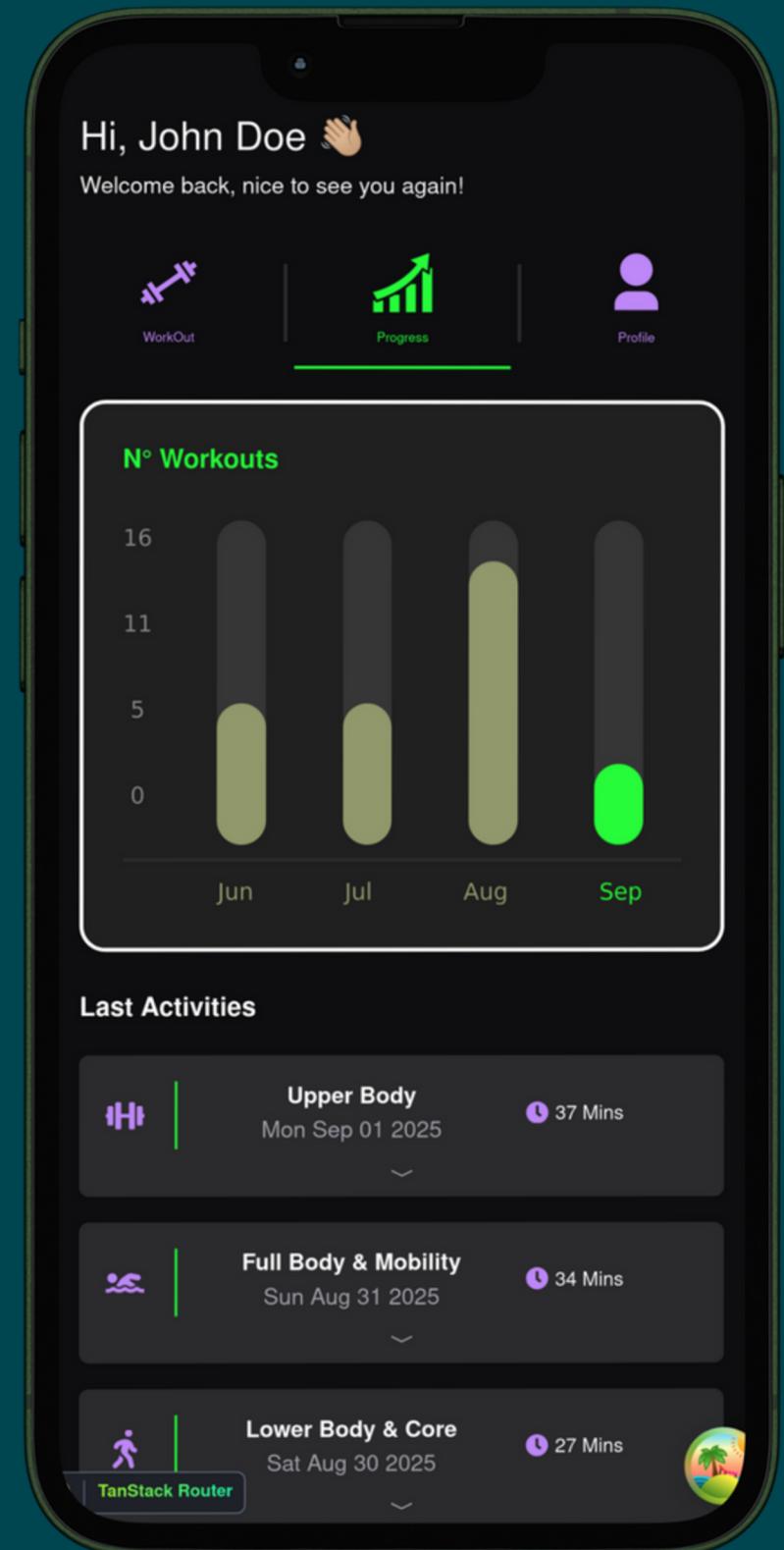
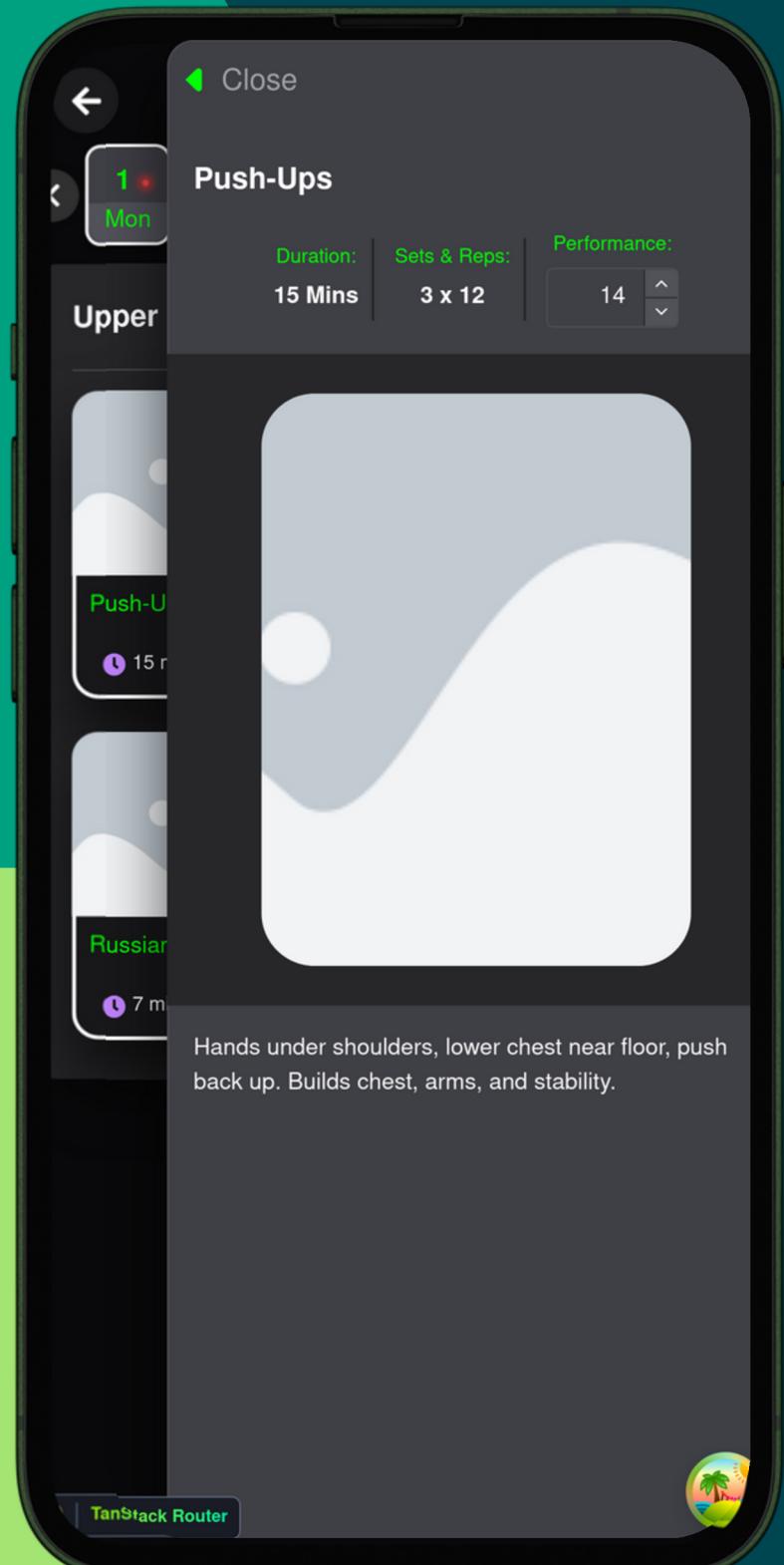
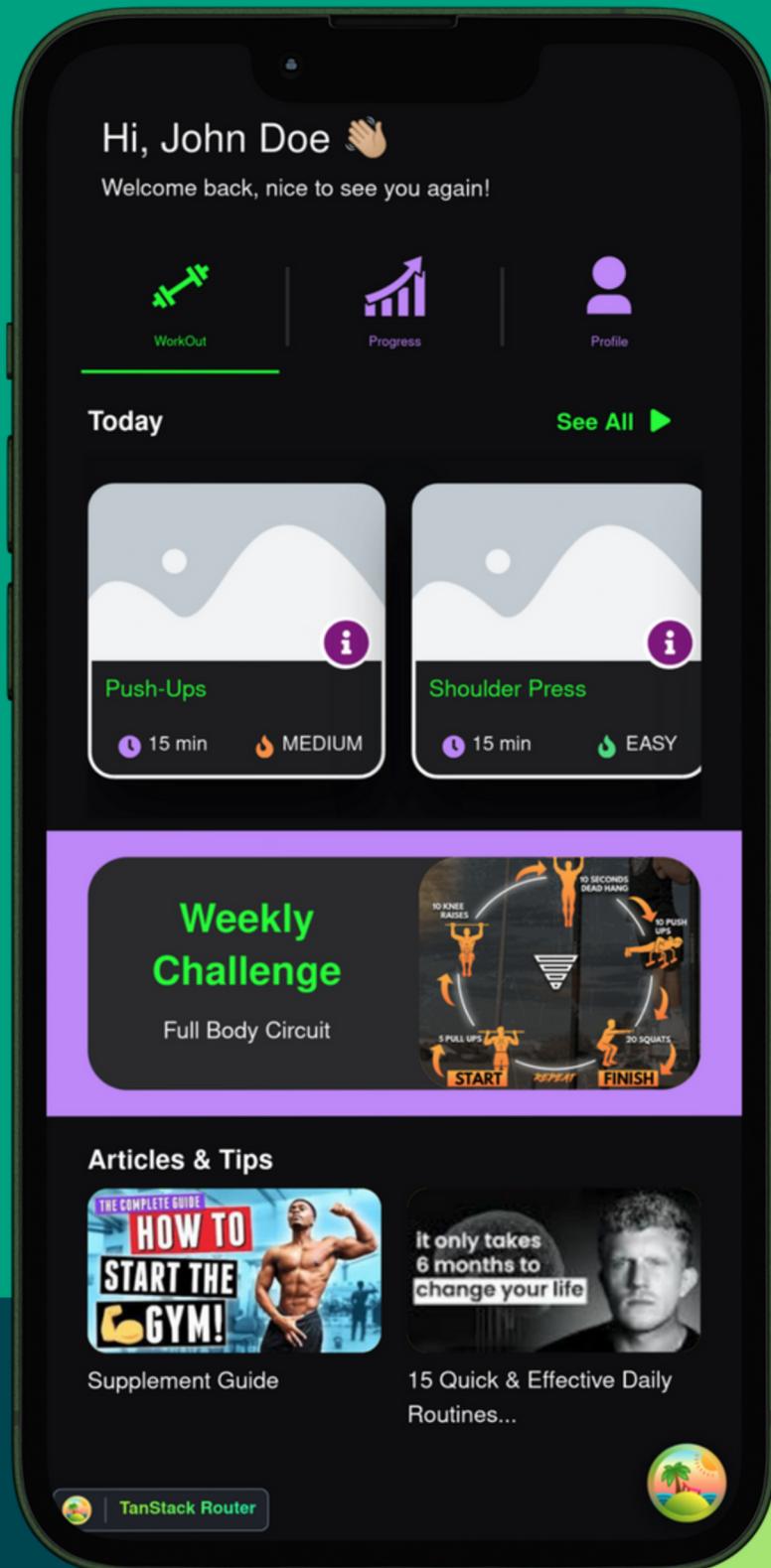
Design



Design



Design





React Query

```
// Update user mutation
const updateMutation = useMutation({
  mutationFn: (data: UserUpdate) =>
    UsersService.updateUserApiV1({ userId, requestBody: data }),
  onSuccess: () => {
    showSuccessToast("Profile updated successfully.")
  },
  onError: (err: ApiError) => {
    handleError(err)
    showErrorToast("Failed to update profile. Please try again.")
  },
  onSettled: () => {
    queryClient.invalidateQueries({ queryKey: ["user", userId] })
  },
})
```

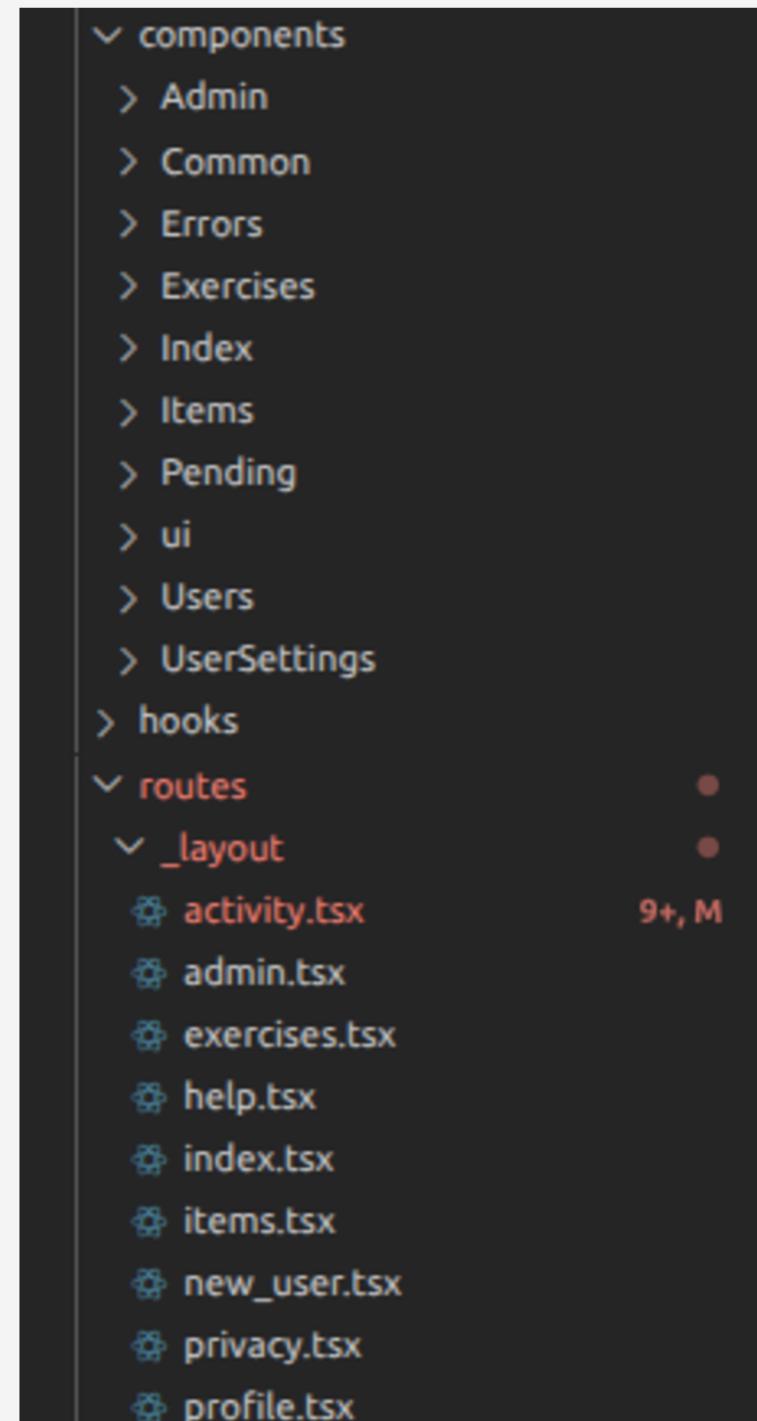
typing dei dati

funzione auto-generata da hey-api

feedback all'utente

gestione degli errori e
della cache

Components & Routes



```
interface ExerciseCardProps {
  exercise: ExercisePublic;
  size?: string;
  onPlay?: (exercise: ExercisePublic) => void;
}

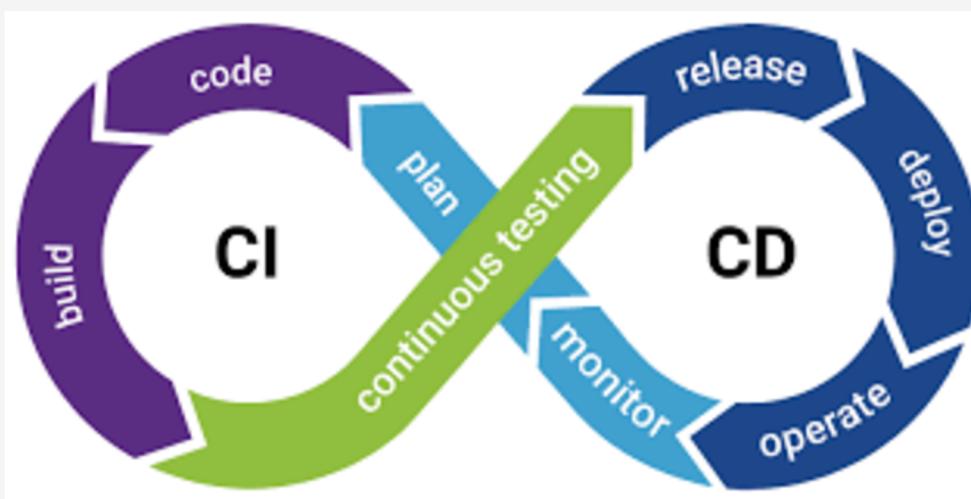
function ExerciseCard({ exercise,
  size = "180px",
  onPlay }: ExerciseCardProps) {
  const [_, setExercise] = useState(exercise);
  const [_, setOnPlay] = useState(onPlay);

  return (
    <Box
      bg="gray.900"
      borderRadius="2xl"
      overflow="hidden"
      boxShadow="2xl"
      position="relative"
      color="white"
    >
```

```
export const Route = createFileRoute("/_layout/user/$userId/profile")({
  component: UserPage,
})

function UserPage() {
  const { userId } = Route useParams<{ userId: string }>()
  const queryClient = useQueryClient()
  const { showSuccessToast, showErrorToast } = useCustomToast()
  const { user: currentUser } = useAuth()
```

Sviluppi futuri



- Deploy



- Gamification



- Data Analysis



- Accesibilità