

Name: Bhatt Priyam

Step 1 — Install Requirements

- `fastapi`
- `uvicorn`
- `sqlalchemy`
- `passlib[bcrypt]`
- `python-jose`

Code:

`database.py`:

```
from sqlalchemy import create_engine
from sqlalchemy.ext.declarative import declarative_base
from sqlalchemy.orm import sessionmaker

DATABASE_URL = "sqlite:///./habits.db"

engine = create_engine(DATABASE_URL,
connect_args={"check_same_thread": False})

SessionLocal = sessionmaker(bind=engine)
Base = declarative_base()
```

`models.py`:

```
from sqlalchemy import Column, Integer, String, Boolean, ForeignKey
from database import Base

class User(Base):
    __tablename__ = "users"

    id = Column(Integer, primary_key=True)
    username = Column(String, unique=True)
```

```
password = Column(String)

class Habit(Base):
    __tablename__ = "habits"

    id = Column(Integer, primary_key=True)
    title = Column(String)
    completed = Column(Boolean, default=False)
    user_id = Column(Integer, ForeignKey("users.id"))
```

schemas.py:

```
from pydantic import BaseModel
```

```
class UserCreate(BaseModel):
    username: str
    password: str
```

```
class HabitCreate(BaseModel):
    title: str
```

```
class HabitOut(BaseModel):
    id: int
    title: str
    completed: bool
```

```
class Config:
    from_attributes = True
```

auth.py:

```
from passlib.context import CryptContext

pwd = CryptContext(schemes=["bcrypt"])

def hash_password(password):
    return pwd.hash(password)

def verify_password(password, hashed):
    return pwd.verify(password, hashed)
```

crud.py:

```
from sqlalchemy.orm import Session
import models
import auth

def create_user(db: Session, username, password):
    hashed = auth.hash_password(password)
    user = models.User(username=username, password=hashed)
    db.add(user)
    db.commit()
    db.refresh(user)
    return user

def create_habit(db: Session, title, user_id):
    habit = models.Habit(title=title, user_id=user_id)
    db.add(habit)
    db.commit()
    db.refresh(habit)
    return habit

def get_habits(db: Session, user_id):
```

```
        return db.query(models.Habit).filter(models.Habit.user_id ==  
        user_id).all()
```

main.py:

```
from fastapi import FastAPI, Depends  
from sqlalchemy.orm import Session
```

```
import models, schemas, crud  
from database import engine, SessionLocal, Base
```

```
Base.metadata.create_all(bind=engine)
```

```
app = FastAPI()
```

```
def get_db():  
    db = SessionLocal()  
    try:  
        yield db  
    finally:  
        db.close()
```

```
@app.post("/register")  
def register(user: schemas.UserCreate, db: Session = Depends(get_db)):  
    return crud.create_user(db, user.username, user.password)
```

```
@app.post("/habit/{user_id}")  
def create_habit(user_id: int, habit: schemas.HabitCreate, db: Session =  
Depends(get_db)):  
    return crud.create_habit(db, habit.title, user_id)
```

```
@app.get("/habits/{user_id}")
def get_habits(user_id: int, db: Session = Depends(get_db)):
    return crud.get_habits(db, user_id)
```

Run Code:

```
uvicorn main:app --reload
```