INTERNHUB

AI-Based Internship and Scholarship Platform
(Diploma Students)

Software Implementation

Academic Year 2025–2026

Contents

1 Project Structure **2**

2 Database Models **3**

2.1 Student Profile Model 3

2.2 Opportunity Model 3

3 Authentication Module **5**

3.1 Registration 5

3.2 Login 8

3.3 Logout 8

4 AI Recommendation Engine **9**

5 Dashboard and Listings **10**

6 Web Scraping Module **11**

7 Conclusion **12**

Chapter 1

Project Structure

```
internhub/
 manage.py
 internhub/
 accounts/
 opportunities/
 ai_engine/
 scraper/
 templates/
 db.sqlite3
```

Chapter 2

Database Models

2.1 Student Profile Model

```
from django.contrib.auth.models import User
from django.db import models

class StudentProfile(models.Model):
    user = models.OneToOneField(User, on_delete=models.CASCADE)
    qualification = models.CharField(max_length=50, default="
       Diploma")
    branch = models.CharField(max_length=100)
    year = models.IntegerField()
    semester = models.IntegerField()
    skills = models.TextField()
    college_name = models.CharField(max_length=200)

    def __str__(self):
        return self.user.username
```

2.2 Opportunity Model

```
from django.db import models

class Opportunity(models.Model):
    TYPE_CHOICES = (
```

```
        ('Internship', 'Internship'),
        ('Scholarship', 'Scholarship'),
    )

    title = models.CharField(max_length=200)
    description = models.TextField()
    eligibility = models.CharField(max_length=200)
    branch = models.CharField(max_length=100)
    skills = models.TextField()
    deadline = models.DateField()
    type = models.CharField(max_length=20, choices=TYPE_CHOICES)
    source = models.CharField(max_length=50, default="Manual")

    def __str__(self):
        return self.title
```

Chapter 3

Authentication Module

3.1 Registration

```
from django.contrib.auth.models import User
from django.shortcuts import render, redirect
from .models import StudentProfile

def register(request):
    if request.method == "POST":
        user = User.objects.create_user(
            username=request.POST['email'],
            email=request.POST['email'],
            password=request.POST['password']
        )

        StudentProfile.objects.create(
            user=user,
            qualification="Diploma",
            branch=request.POST['branch'],
            year=request.POST['year'],
            semester=request.POST['semester'],
            skills=request.POST['skills'],
            college_n\documentclass[12pt,a4paper]{report}

\usepackage{geometry}
\geometry{margin=1in}
```

```
\usepackage{setspace}
\usepackage{graphicx}
\usepackage{hyperref}
\usepackage{listings}
\usepackage{xcolor}

\definecolor{codegray}{rgb}{0.95,0.95,0.95}

\lstset{
    backgroundcolor=\color{codegray},
    basicstyle=\ttfamily\small,
    frame=single,
    breaklines=true
}

\title{\textbf{InternHub}\\
Internship \& Scholarship Finder Platform}
\author{Project by Diploma Students}
\date{}

\begin{document}

\maketitle
\onehalfspacing

\tableofcontents
\newpage

%--------------------------------------------------

\chapter{Introduction}
InternHub is a web-based platform designed to help diploma
   students find internships and scholarships.
\documentclass[12pt,a4paper]{report}

\usepackage{geometry}
\geometry{margin=1in}
```

```
\usepackage{setspace}
\usepackage{graphicx}
\usepackage{hyperref}
\usepackage{listings}
\usepackage{xcolor}

\definecolor{codegray}{rgb}{0.95,0.95,0.95}

\lstset{
    backgroundcolor=\color{codegray},
    basicstyle=\ttfamily\small,
    frame=single,
    breaklines=true
}

\title{\textbf{InternHub}\\
Internship \& Scholarship Finder Platform}
\author{Project by Diploma Students}
\date{}

\begin{document}

\maketitle
\onehalfspacing

\tableofcontents
\newpage

%------------------------------------------------

\chapter{Introduction}
InternHub is a web-based platform designed to help diploma
   students find internships and scholarships.
ame=request.POST['college']
        )
        return redirect('login')

```

```
96      return render(request, 'register.html')
```

3.2 Login

```
1  from django.contrib.auth import authenticate, login
2
3  def login_user(request):
4      if request.method == "POST":
5          user = authenticate(
6              username=request.POST['email'],
7              password=request.POST['password']
8          )
9          if user:
10             login(request, user)
11             return redirect('dashboard')
12     return render(request, 'login.html')
```

3.3 Logout

```
1  from django.contrib.auth import logout
2
3  def logout_user(request):
4      logout(request)
5      return redirect('login')
```

Chapter 4

AI Recommendation Engine

```
def recommend(student, opportunities):
    ranked = []

    for opp in opportunities:
        score = 0

        if "Diploma" in opp.eligibility:
            score += 50

        if student.branch.lower() in opp.branch.lower():
            score += 20

        for skill in student.skills.lower().split(','):
            if skill.strip() in opp.skills.lower():
                score += 10

        ranked.append((score, opp))

    ranked.sort(reverse=True, key=lambda x: x[0])
    return [opp for score, opp in ranked[:5]]
```

Chapter 5

Dashboard and Listings

```python
from django.shortcuts import render
from accounts.models import StudentProfile
from .models import Opportunity
from ai_engine.recommender import recommend

def dashboard(request):
    student = StudentProfile.objects.get(user=request.user)
    data = Opportunity.objects.filter(
        eligibility__icontains="Diploma"
    )
    recommendations = recommend(student, data)

    return render(request, "dashboard.html", {
        "recommendations": recommendations
    })
```

Chapter 6

Web Scraping Module

```
import requests
from bs4 import BeautifulSoup
from opportunities.models import Opportunity

def scrape():
    url = "https://example.com/diploma-internships"
    page = requests.get(url)
    soup = BeautifulSoup(page.text, 'html.parser')

    for job in soup.find_all("div", class_="job"):
        title = job.find("h2").text
        eligibility = job.find("span").text

        if "Diploma" in eligibility:
            Opportunity.objects.create(
                title=title,
                description=job.text,
                eligibility=eligibility,
                branch="Any",
                skills="Basic Skills",
                deadline="2026-12-31",
                type="Internship",
                source="Scraped"
            )
```

Chapter 7

Conclusion

This chapter presented the complete software implementation of INTERNHUB using Django, including authentication, AI-based recommendations, and automated web scraping tailored for Diploma students.