

SickleSmart - MVP Complete

“Your partner in Sickle Cell care: Empowering lives, one cell at a time.”

PROGRESS

Backend Development and Enhanced User Experience! 😁

This week, I would confidently rate my progress at 8. I dedicated my focus to advancing the backend functionality of the project. The primary achievement is the successful implementation of user authentication, enabling seamless user registration and login capabilities on the website.

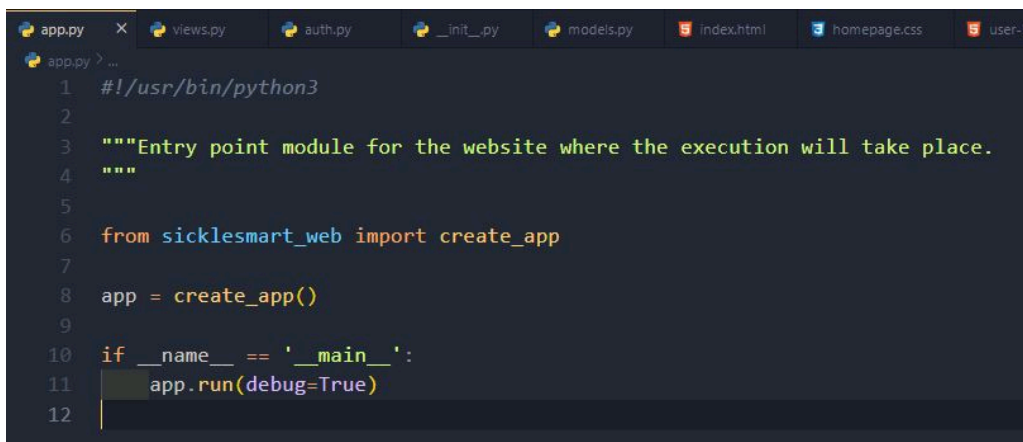
In the backend, I utilized Flask's hash password mechanisms to ensure robust password security. The user database has been efficiently structured, storing crucial information for each user. Notably, serious checks have been implemented to verify user passwords, enhancing the overall security of the application.

On the frontend, I devoted time to refining the user interface. The register and login pages have been aesthetically revamped, moving away from the conventional and ensuring a more engaging and user-friendly experience.

While substantial strides have been made, there's an exciting feature in the pipeline. I'm working towards integrating a functionality where users can receive personalized diet and health modifications based on their current health status. This promises to add a valuable dimension to the user experience and is something to anticipate in the upcoming stages of development.

For a glimpse into the code structure, here are snippets from key files:

1. Entry Point (app.py):



```
1  #!/usr/bin/python3
2
3  """Entry point module for the website where the execution will take place.
4  """
5
6  from sicklesmart_web import create_app
7
8  app = create_app()
9
10 if __name__ == '__main__':
11     app.run(debug=True)
12
```

2. Views Blueprint (views.py):

```
@views.route('/user-logged_in', methods=['GET', 'POST'])
@login_required
# to be added to the page route that should be displayed when
# users have successfully registered or logged in.
def user_logged_in():
    if request.method == 'POST':
        healthStatus = request.form.get('healthStatus')

        if len(healthStatus) < 1:
            flash('Details 😊, how do you actually feel?', category='error')
        else:
            new_status = HealthStatus(symptoms=healthStatus, user_id=current_user.id)
            db.session.add(new_status)
            db.session.commit()
            flash('Health tracked successfully! Remember to drink more water. 😊',
                  category='success')

    return render_template('user-logged_in.html', user=current_user)
```

3. Authentication Blueprint (auth.py):

```
12 auth = Blueprint('auth', __name__)
13
14
15 @auth.route('/login', methods=['GET', 'POST'])
16 def login():
17     if request.method == 'POST':
18         email = request.form.get('email')
19         password = request.form.get('password')
20
21         user = User.query.filter_by(email=email).first()
22         if user:
23             if check_password_hash(user.password, password):
24                 flash('Logged in successfully!', category='success')
25                 login_user(user, remember=True)
26                 return redirect(url_for('views.user_logged_in'))
27             else:
28                 flash('You dey wyn me? E no correct jhur', category='error')
29         else:
30             flash('Email does not exist - I no sabi you before',
31                   category='error')
32
33     return render_template("login.html", user=current_user)
34
```

4. Database Module (models.py):

```
20 class User(db.Model, UserMixin):
21     id = db.Column(db.Integer, primary_key=True)
22     email = db.Column(db.String(150), unique=True)
23     password = db.Column(db.String(150))
24     user_name = db.Column(db.String(150))
25     healthStatus = db.relationship('HealthStatus')
26
```

CHALLENGES

Technical Challenges - Empowering User Health Tracking☺:

Currently, the user's method of tracking health on the website involves expressing their well-being through short notes or descriptions. While this unconventional approach allows users to document their feelings, there's a challenge to enhance this process. The hurdle lies in providing users with the capability to manage their health entries effectively.

A significant challenge I'm tackling is the implementation of a feature that allows users to delete specific entries. The objective is to empower users to maintain a dynamic and organized health log, removing entries once they've addressed the related concerns or received necessary treatments. This functionality is crucial for users who wish to streamline their health records and have a clear overview of their progress over time.

The complexity arises in creating a seamless and secure process for users to selectively delete entries from the database. Balancing user-friendliness with data integrity is paramount, ensuring that users can effortlessly manage their health records without compromising the accuracy and completeness of their overall health history.

Addressing this challenge involves careful consideration of user experience design and backend functionality. As I delve into this technical hurdle, my aim is to provide users with a refined and intuitive interface, empowering them to take control of their health tracking journey on the website. Stay tuned for further updates as I navigate and conquer this intricate aspect of the project.

Non-Technical Challenges:

Beyond the realm of code and databases, non-technical challenges have presented themselves, impacting the project's pace. A significant obstacle I've encountered is the intermittent availability of electricity. This external factor has introduced considerable disruptions to my workflow, leading to extended periods without commits to the repository due to a drained laptop. The unreliable power supply has proven to be a formidable adversary, demanding adaptability and resilience.

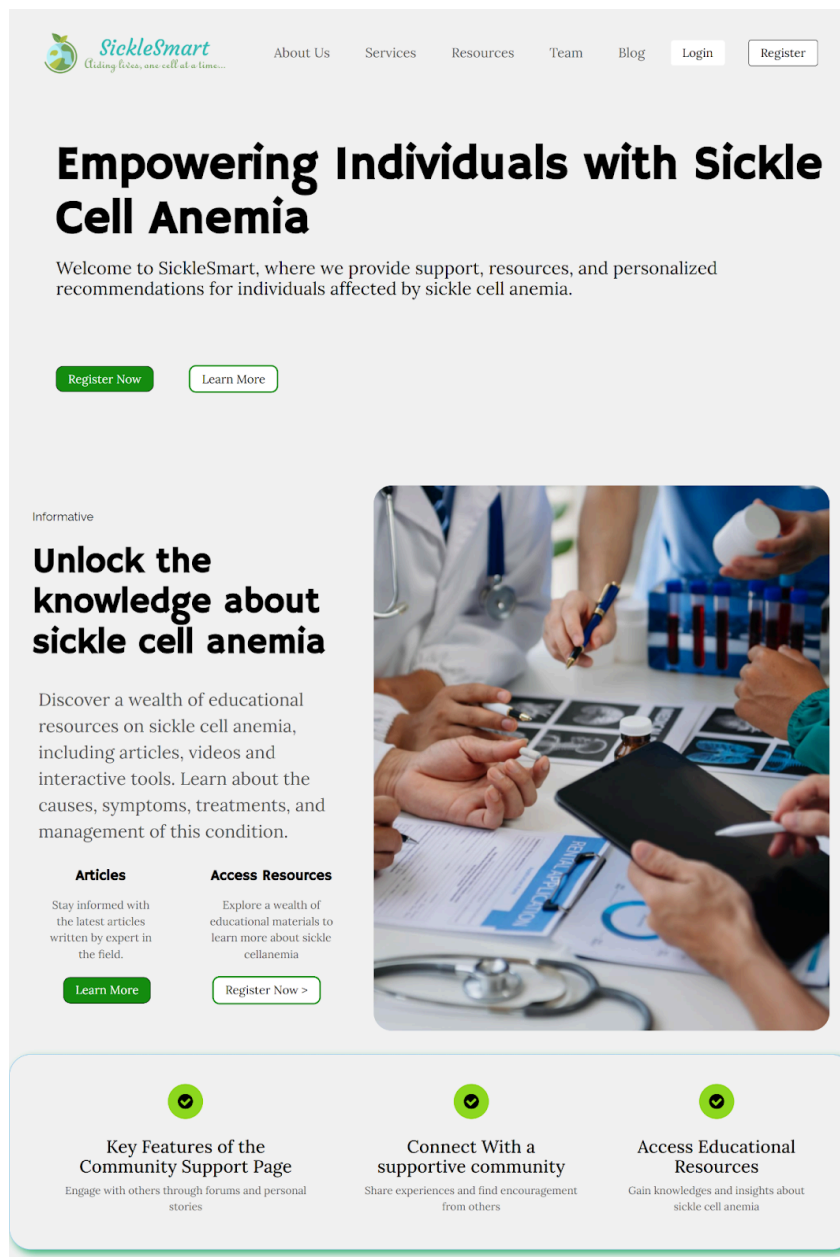
Moreover, the work environment's impact on morale has become evident. Creating a conducive space for coding is crucial, and the absence of such an environment has posed an additional non-technical challenge. A more stable and comfortable

workspace is essential for fostering focus and creativity, elements vital to overcoming coding hurdles effectively.


Addressing these non-technical challenges involves exploring solutions for power backup or alternative workspaces, ensuring a more consistent and productive development journey. Despite these hurdles, the commitment to the project remains unwavering, and strategies are being devised to mitigate the impact of external factors on both productivity and morale.

SHARE SCREENSHOTS

1. Preview of the landing page:



2. Overview of the Logged-in page for registered user:

[About Us](#)[Why? 🧑🏾](#)[Effects 🩺](#)[Resources 📖](#)[Blog](#)[Dashboard](#)[Logout](#)


Welcome to SickleSmart 🧑🏾, David!

Sickle cell disease is a hereditary blood disorder characterized by the presence of abnormal hemoglobin, causing red blood cells to take on a sickle shape. This unique shape can lead to a variety of complications, affecting both physical and overall well-being. This can lead to a range of health complications, including pain crises, organ damage, and an increased risk of infections.

Understanding Sickle Cell Disease

Sickle hemoglobin (HbS) differs from normal hemoglobin (HbA) in its ability to form long, rigid rods within the red blood cells, especially in low-oxygen conditions. This process is called sickling, and it contributes to the following aspects of the disease:

Pain Crises: The sickle-shaped cells can block blood flow, leading to episodes of intense pain, known as pain crises.	Organ Damage: Sickle cells can cause damage to various organs, affecting their normal functioning.
Increased Infection Risk: Sickle cells have a shorter lifespan, leading to a reduced ability to fight infections.	Anemia: The breakdown of sickle cells can result in a shortage of red blood cells, leading to anemia.



- [Ayomide Kayode](#) 🙌😊

