Project webbramverk

Amir Ramic & Amer Ahmed

Backlog 17/1 -22 (Startup of project)

Type: Creation of Flask project
Description: Creation and discussion of the flask project which would be executed in Pycharm, discussion download of necessary plug-ins such as Flask, flask-wtf & wtforms.
Estimated time: 60min
Actual time: 60min

Type: Database Design
Description: Design and discussion of the most suitable database for this project. Conclusion was to create an SQL relational database model with necessary tables and relations for the startup of the project, enabling us to create a starting point for furthermore work
Estimated time: 60 min
Actual time: 60 min

Type: User table
Description: Creation of a User table in pycharm with following attributes, User_id, Username, Email, User_image, Password & Posts.
Estimated time: 20min
Actual time: 10 min

Type: Post table
Description: Creation of a Post table in pycharm with following attributes, Post_id, Title, Date, Content & (ForeignKey User_id) This table is set to an ‘One to many’ relation with the User table, to create a communication between these two tables.
Estimated time: 20 min
Actual time: 10 min

Type: Form and validation
Description: Creation and implementation of forms using wtforms and validation of user input. To minimize errors such as, no email, wrong email, no password & 2 factor authentication of password. Also to create a visualization of a fill-in form.
Estimated time: 60 min
Actual time: 45 min

Type: Create forms Login and Registration
Description: To create both fill-in forms on separate routes with errorhandling of following problems, no email input, wrong email, no password input, wrong password, 2 factor authentication of password. Also to create the initial design of the fill-in forms.
Estimated time: 60 min
Actual time: 80 min

Type: Class Registration Form
Description: Creation of a class in python enabling us to create a fill-in form. This is to minimize errors of email input and password input with a 2 factor authentication of the password.
Estimated time: 45 min
Actual time: 60 min

Type: Class Login form
Description: Creation of a class in python enabling us to create a login form where a user can sign in. To create a errorhandling of wrong password & email input.
Estimated time: 45 min
Actual time: 45 min

Sprint planning:

Name of spring: Startup of project

Description: Startup of project and verification of the necessary tools for the creation of a flask project with a database communication. The program should be able to recognize wrong input from users, able to sign in and to register a user on a localhost website but without the ability of storing data in a database since the implementation and creation of a database has not yet been done.

Start date: 19/1-22
End date: 21/1-22

Backlog :

- **Creation of flask project**
  - **Database design**
    - **User table**
    - **Post table**
  - **Form and validation**
- **Create forms login and registration**
  - **Class RegistrationForm**
  - **Class LoginForm**

Daily standup:
21/1-22
Attendees: Amir Ramic and Amer Ahmed

Since 19/1-22 we have resolved these issues:

Creation of flask project
Database Design
User table
Post table

We plan to work on these issues:
Form and validation
Create forms login and registration
Class RegistrationForm
Class LoginForm

We have not noticed any need for extra resources for the continuation of this project thus leaving us with the conclusion that everything should work out as planned accordingly.

Daily standup:
21/1 -22
Attendees: Amir Ramic and Amer Ahmed

Since 19/1-22 we have resolved these issues:

Form and validation
Create forms login and registration
Class RegistrationForm
Class LoginForm

SPRINT REVIEW:

All issues were done during the planned sprint.
Since this was our first sprint we tried to feel how everything would work out, our conclusion of this sprint is that we planned and structured it very well which gave us the impression of being on time with everything.
Furthermore we will plan the sprints with the same method which is a logical approach to the upcoming issues.
New issues have emerged in the backlog after the sprint review.

Backlog 21/1-22:

Type: Create Database
Description: Creation of the database in SQLAlchemy to enable the storage of data
Estimated time: 20 min
Actual time: 15 min

Type: Create User class
Description: Creation of the User class which should be connected to the database Users table. Enabling storage of user info such as: Username, Email, User_image Password & Posts.
Estimated time: 10 min
Actual time: 10 min

Type: Create Post class
Description: Creation of the Post class which should be connected to the database Post table. Enabling storage of post info such as: Title, Date, Content & (ForeignKey User_id)
Estimated time: 10min
Actual time: 10 min

Type: SQLAlchemy database communication
Description: Startup and linkup with the database in SQLAlchemy to Pycharm.
Estimated time: 10 min
Actual time: 10 min

Type: User model creation
Description: User model creation in Pycharm to clarify what kind of attributes and relations the User model has. Such as Nullable or DataRequired
Estimated time: 5min
Actual time: 5 min

Type: Post model creation
Description: Post model creation in Pycharm to clarify what kind of attributes and relations the Post model has. Such as Nullable or DataRequired
Estimated time: 5 min
Actual time: 5 min

Type: Project package organization
Description: Organization of the code into packages to make it more understandable and readable, the goal is also to avoid merge conflicts when committing into Github thus making it possible to work in two separate packages without conflicts.
Estimated time: 20 min
Actual time: 30 min

Type: Create hash password method
Description: To create a method where the password that is stored in the database is hashed and not in danger of getting exposed for intruders.
Estimated time: 60 min
Actual time: 45 min

Type: Implement hash method into RegistrationForm
Description: Implementation of the hash password method into the RegistrationForm to protect user profiles and passwords from being exposed to intruders when stored in the database. Password is not visually seen when being input into the form, dots replace the text thus the 2 factor authentication which helps the user know that the password has been input correctly 2 times into the form.
Estimated time: 90 min
Actual time: 70 min

Type: Implement hash method into LoginForm
Description: Implementation of the hash password method into the LoginForm where the program recognizes the correct password when trying to login, password is stored in a hash value and the password is not visually seen when input is being made into the form, protection with dots instead of text.
Estimated time: 90 min
Actual time: 90 min

Type: UserMixIn LoginManager
Description: Usage of the UserMixIn, creation and implementation of the method Loginmanager into the program which trough flask-login UserMixIn helps the LoginManager authenticate that the user has provided valid input such as Email, Password, this is to easier check if the User is active in the database. Enabling us to use the Is_authenticated method.
Estimated time: 60 min
Actual time: 70 min

Type: User Authentication
Description: Authentication if the user has provided valid input
Estimated time: 20min
Actual time: 30 min

Type: User account
Description: Creation of the first user account with stored data in the database. To test if the database responds when making input from a localhost website.
Estimated time: 10 min
Actual time: 5 min

Type: Username validation
Description: Validation of the given username. To make sure the username is within the given parameters.
Estimated time: 20 min
Actual time: 20 min

Type: Email validation
Description: Validation of the input email. To make sure that the user has input a valid email with the @ sign. If a wrong email is present, an error will occur.
Estimated time: 20min
Actual time: 20 min

Sprint planning:

Name of spring: Second sprint

Description: Further builds on the project with a connected database, focus on security with hashed password method and validation of the user input such as Username, Email, Password.
The program should be able to store User information into the database and the password should be protected by a hashed value.

Start date: 24/1-22
End date: 28/1-22

Backlog :

- **Create Database**
- **Create User class**
- **Create Post class**
- **SQLAlchemy Database communication**
- **User model creation**
- **Post model creation**
- **Project package organization**
- **Create Hash password method**
- **Implement hash password method into RegistrationForm**
- **Implement hash password method into LoginForm**
- **UserMixIn Loginmanager**
- **User Authentication**
- **User account**
- **Username validation**
- **Email validation**

Daily standup:
24/1 -22
Attendees: Amir Ramic and Amer Ahmed

Since 19/1-22 we have resolved these issues:

Form and validation
Create forms login and registration
Class RegistrationForm
Class LoginForm

We plan to work on these issues:

Create Database
Create User class
Create post class
SQLAlchemy Database communication
User model creation
Post model creation
Project package organization
Project package organization
Create Hash password method
Implement hash password method into RegistrationForm
Implement hash password method into LoginForm
UserMixIn Loginmanager

We have not noticed any change in the need for resources which could stop this project from proceeding.
The project work continues as planned.

Daily standup:
28/1 -22
Attendees: Amir Ramic and Amer Ahmed

Since 24/1-22 we have resolved these issues:

Create Database
Create User class
Create post class
SQLAlchemy Database communication
User model creation
Post model creation
Project package organization
Project package organization
Create Hash password method
Implement hash password method into RegistrationForm
Implement hash password method into LoginForm
UserMixIn Loginmanager

We plan to work on these issues:

User Authentication
User account
Username validation
Email validation

Daily standup:
31/1 -22
Attendees: Amir Ramic and Amer Ahmed

Since 28/1-22 we have resolved these issues:

User Authentication
User account
Username validation
Email validation

SPRINT REVIEW (Second Sprint):

All issues were done during the planned sprint.
In our second sprint we added more issues than last time because the issues we were facing this time were not as big as in the first sprint.
We planned our sprint as we agreed, with a logical approach thus making it easy to follow and stress free.
We agreed that this is the optimal way of planning the sprints during this project and we will continue doing it.
New issues have emerged in the backlog and are added as well.

Backlog 31/1-22:

Type: User Authentication
Description: To make sure that the logged in user has limited access throughout the web-page. Also to make sure that certain functions only appear if the User is authenticated by the web-page such as only if logged in show functions log-out on navigation bar.
Estimated time: 180min
Actual time: 200 min

Type: User account
Description: To make sure that the user is able to create an user account and that the information is stored in the database and to make sure that the web-page is able to communicate with the database while storing information in it.
Estimated time: 30 min
Actual time: 30 min

Type: Username validation
Description: To create a validation point where the User is asked to input a Username which would be shown on the web-page, the User cannot exist without an username.
Estimated time: 30 min
Actual time: 25 min

Type: Email validation
Description: To create a validation point, to make sure that the User has input a valid email which will be stored in the database.
Estimated time: 30 min
Actual time: 15 min

Type: User account profile
Description: To create a profile picture function for the existing users. With this function the user should be able to upload a profile picture which is shown on the web-page.
Estimated time: 60 min
Actual time: 70 min

Type: Account posts
Description: To give the possibility of creating posts on the web-page through a signed in account, only users that are signed in have the option of creating a new post in the sign bar.
Estimated time: 60 min
Actual time: 60 min

Type: Update account
Description: To create an option where the signed in user is able to update the account with new email or new username.
Estimated time: 60 min
Actual time: 80 min

Type: Update post
Description: To create an option where the signed in user is able to update or modify their post with new text or a new headline.
Estimated time: 60 min
Actual time: 60 min

Type: Delete post
Description: To create an option where the signed in user is able to delete their own posts.
Estimated time: 30 min
Actual time: 45 min

Type: Post Pagination
Description: To create a function on the web-page where the posts that are shown are in a sequenced order, limited by 2 posts per page and to create pages where the other posts are available. This is to prevent scrolling on the page.
Estimated time: 60 min
Actual time: 70 min

Type: Forget password, reset password
Description: To create an option for the user if the password needs to be reset, this creates an email with a hashed token link that is valid for 30 min, after expiration the token is invalid. The token email is sent to the stored user email from the database.
Estimated time: 60 min
Actual time: 90 min

Type: Search function
Description: To create a function where the user has the possibility of searching for a specific post on the web-page.
Estimated time: 45 min
Actual time: 45 min

Type: Error handling (404, 403, 500)
Description: Added error handling into code where if the user encounters one of the following errors (404, 403, 500) the web-page does not crash and is still usable.
Estimated time: 60 min
Actual time: 60 min

Sprint planning:

Name of spring: Third sprint

Description: Further builds on project involving ability to create real user with functioning log in, functioning reset password method, ability to upload profile picture, ability to create, modify and delete posts on page.
The project web-page should be able to paginate posts into different pages, the user should be able to search for a specific post and the web-page should not crash when encountering one of the following errors (404,403,500).

Start date: 31/1-22
End date: 4/2-22

Backlog :

- **User authentication**
  - **User account**
- **Username validation**
  - **Email validation**
- **User account profile**
  - **Account posts**
  - **Update account**
  - **Update posts**
  - **Delete posts**
- **Post pagination**
- **Forget password, reset password**

- **Search function**
- **Error handling (404,403,500)**

Daily standup:
31/1 -22
Attendees: Amir Ramic and Amer Ahmed

Since 24/1-22 we have resolved these issues:

User Authentication
User account
Username validation
Email validation

We plan to work on these issues:
User authentication
User account
Username validation
Email validation
User account profile
Account posts
Update account
Update posts
Delete posts

Daily standup:
2/2 -22
Attendees: Amir Ramic and Amer Ahmed

Since 31/1-22 we have resolved these issues:

User authentication
User account
Username validation
Email validation
User account profile
Account posts
Update account

Update posts
Delete posts

We plan to work on these issues:

Post pagination
Forget password, reset password
Search function
Error handling (404,403,500)

Daily standup:
4/2 -22
Attendees: Amir Ramic and Amer Ahmed

Since 31/1-22 we have resolved these issues:

Post pagination
Forget password, reset password
Search function
Error handling (404,403,500)

SPRINT REVIEW (Third Sprint):
All of the planned issues were resolved during this sprint.
The largest issue we encountered was the user account issue, we knew already from the start that this would be a time consuming part of the project because it involved all parts of the project. This is also why we chose to work on this issue at this stage of the project, just because we needed our web-page to respond with the database, we needed our RegistrationForm and LoginForm methods to work and we needed our password to be stored in a hash value prior to creating a real user just out of safety aspects.
We were pleased by the outcome of this sprint and we created new items in the backlog that should be worked on.

Backlog 4/2-22:

Type: Blueprint
Description: Organizing code into blueprints, to make the code more accessible. Usage and import of flask blueprint, creation of python packages.
Estimated time: 90 min
Actual time: 90 min

Type: Creation of database, docker container
Description: To create a docker container where the database will store the information.
Estimated time: 90 min
Actual time: 45 min

Type: Creation of UML model
Description: To clarify how the project is modeled and how the packages communicate with each other.
Estimated time: 180 min
Actual time: 360 min

Type: Project deployment
Description: Deployment of project, the project is finished in this stage. Demonstration is held
Estimated time: 30 min
Actual time: 30 min

Sprint planning:

Name of spring: Completion sprint

Description: Last works on the project such as organizing code into blueprint packages, creation of a docker container where information will be stored. Lastly, creation of an UML model where clarity between the package communications will be shown.
The last backlog item is the “Project deployment”, which is the actual demonstration of the finished project.

Start date: 7/2-22
End date: 16/2-22

Backlog :

- **Blueprint**
- **Creation of database, docker container**
- **Creation of UML model**
- **Project deployment**

Daily standup:
7/2 -22
Attendees: Amir Ramic and Amer Ahmed

Since 4/2-22 we have resolved these issues:

Post pagination
Forget password, reset password
Search function
Error handling (404,403,500)
We plan to work on these issues:

Blueprint

Daily standup:
9/2 -22
Attendees: Amir Ramic and Amer Ahmed

Since 7/2-22 we have resolved these issues:

Blueprint

We plan to work on these issues:

Creation of database, docker container

Daily standup:
11/2 -22
Attendees: Amir Ramic and Amer Ahmed

Since 7/2-22 we have resolved these issues:

Creation of database, docker container

We plan to work on these issues:

Creation of UML model

Daily standup:
14/2 -22
Attendees: Amir Ramic and Amer Ahmed

Since 11/2-22 we have resolved these issues:

Creation of UML model

SPRINT REVIEW (Completion sprint):

Since this was the last sprint of the project, we planned to work over a longer period of time than in the last sprint.
The most time consuming backlog item was the UML creation which gave us the conclusion that we actually could have created the UML during the progression of the project and not all at once in the end, we were not in danger of missing the deadline date but unnecessary stress could have been avoided, we did not know how time consuming it could be to create an UML model especially if it is your first time.
The last issue (Project Deployment) is actually the demonstration of the project and is not considered as a real coding issue, that is the reason why it is not included in the final day of the sprint .
With that said, the project and the last sprint was successfully brought to an end, we were pleased by the outcome of the project, the communication in the group and the planning of each sprint.
Since there will not be any furthermore sprint this project is officially completed.