**Development Model**

Agile model is most appropriate for my project for a number of reasons:

Ability to change my code and improve it.

Ability to iterate, make small increments and testing.

Flexibility to incorporate changes even in the later stages of the code.

Continues improvement of the code.

Full control of the development, as I am working by myself this model suits me best because I can control the flow and way I work around with my project as I have no other team members to depend/rely on.

**Application Requirements**

**Functional Requirements**

1. User Registration and Login: The website has a user registration page (sign up) for users that aren’t registered and a login page for users that have been registered. These users can only sign up and register with their university emails.
2. Profile Creation: There is an edit profile feature where the user can customize their name, profile picture, contact information, major, interests, and skills.
3. Forum or Discussion Board: The website has a forum/discussion board where students share information, post questions, start discussions related to their academic or extracurricular activities etc.
4. User-generated Content: The website has a feature that allows users to create and post their own content, such as questions, information articles, blog posts, interests, or academic pursuits. This feature fosters a community spirit and encourages knowledge-sharing among students.
5. Tagging and Flair Options: The website has tagging and flair options when users create a post that allow them to categorize their posts and make them more discoverable for others. Tags have to be descriptive keywords that represent the context of the content on the post. Flairs are labels which indicate the post's topic or category which is related to the department, course program etc. These features make it easier for users to identify pertinent content and enable a more structured and efficient browsing experience.
6. Search Functionality: The website has a search bar that allows users to search for specific information or content within the website. Tags and flairs will help to make this search functionality more efficient and useful to the users, as in this way they can find specific posts, documents, questions within the community.
7. Like and comments: The website provides like and comments system where users can interact with posts by liking them or commenting their opinions, answers etc.
8. Notifications: The website provides a notification system that allows students to receive updates on record time related to people that interact with their posts.
9. Group chats: Students are able to join already existing groups for each small community inside the web application, as well as create their own group chats with fewer people in.
10. Document Sharing: The website has a document sharing system where students can upload in pdf format materials such as lecture notes and study guides.
11. Bookmarks: The website allows users to save/bookmark posts and other content that they want to refer to later. Helps users to find and access the saved content easily, without having to search for it again, this makes the usage more efficient.
12. News and Updates : The website has a news and updates section that keeps students updated related to policy changes, updates on the website and other issues that might concern them.
13. Messaging System: The website has a messaging system which allows students to privately communicate with one another.
14. Feedback or Suggestions: The website has a feedback and suggestions section which receives students’ feedback and suggestions to improve the website's functionality and features as well as provide complaints about users that have misbehaved.
15. Report Post Option: The website has a report post option that allows users to flag inappropriate or offensive content or behavior. This way the community can maintain a safe and respectful environment and prevent any form of harassment or abuse.
16. Admin Page for Moderators: The website has an admin page that only designated moderations can access. This page provides moderators with necessar tools to manage and monitor the community and its content. Some functionalities on the admin page are:

o   Reviewing reported content by users and taking appropriate action, such as deleting or suspending users who violate community guidelines.

o   Managing user accounts, disabling or deleting inactive or problematic accounts in the community.

o   Creating and managing community guidelines, policies, and rules, and updating them as necessary to ensure a safe and respectful community environment.

o   Communicating with users, such as sending announcements or responding to feedback or support requests.

**Non- Functional Requirements**

1. Usability: The web application should be designed so that it is easy to use and understand, with clear navigation and an organized content.
2. Performance: The web application should be able to handle traffic and a large number of users without slowdowns, errors or bugs. It should be responsive for all types of browsers and devices. It should be able to load quickly and efficiently the pages.
3. Security: The web application should assure the security and protection of user data and prevent unauthorized access. Sensitive data and passwords should be encrypted. Users should be given security options such as email notifications and 2 FA login. The web application should constantly assure safety for its users with regular updates.
4. Reliability: The web application should be available at all times with no disruptions and minimal downtime. If any outages or errors occur it should have a quick recovery process.
5. Scalability: The web application should be able to scale up or down and accommodate the increase of amounts of data, traffic as well as work volumes and user growth.
6. Maintainability: The web application should be easy to update and maintain through easy to modify and test code.
7. Compatibility: The web application should be responsive and compatible in different browsers, devices and platforms.
8. Localization: The web application should be easy to use and accommodate users from different regions of the country by using the unified English language.
9. Availability: The web application should be available at all times with minimal downtime. Good cloud-based infrastructure and safe servers that help to monitor and ensure that issues are handled on time.
10. Portability: The web application should be portable between different environments such as development, testing and production.  This could require utilizing technologies such as containerization or virtualization, along with tools for configuration management, to maintain uniformity across diverse environments.

**User requirements**

**Function requirements**

1. Users are able to create an account and login to the website through their student email.
2. Users are able to edit their profile by customizing their name, profile picture, contact information, major, interests and skills.
3. Users are able to join forums, ask questions, share information etc.
4. Users are able to create and post their own content such as questions, blog posts, articles, interests, academic pursuits.
5. Users are able to tag and categorize their posts so that they fit certain mini communities and make discoverability easier.
6. Users are able to search for specific information in the search bar through keywords.
7. Users are able to receive and view their notifications related to interactions they have gotten on their posts.
8. Users are able to join or create groups based on their interests or majors and chat in them.
9. Users are able to upload and share academic materials, also download documents uploaded by other students.
10. Users are able to save and bookmark posts in order to find them easily without have to search again.
11. Users are able to access the news and updates section where they can get update by the developer of the community.
12. Users are able to privately communicate with other students on direct messages.
13. Users are able to like and share posts on their page.
14. Users are able to submit feedback to the community, raise complaints, make suggestions etc.
15. Users are able to report inappropriate content or behavior through the report option on the posts.

**Nonfunctional requirements**

1. Users are able to use the community easily, with clear navigation and organized content.
2. Users expect a fast, responsive website that can handle a huge number of users without slowdowns, errors or bugs.
3. Users expect their personal data and information to be protected and secured through encryption and secure login processes. 2FA option available, and password protection.
4. Users expect little to no downtime or disruptions so that the community operates at all times.
5. Users expect the web application to be compatible with different browsers and devices as well as responsive on different platforms.