SE 216 – SOFTWARE PROJECT MANAGEMENT
SOFTWARE PROCESS MODEL DOCUMENT

PROJECT NAME: MediShare

GROUP MEMBERS: Bartu ERDEM, Batuhan CAN, Erkan Efe YEŞİLÇİMEN, Emin Buğra Aksoy, Hüseyin Atacan AKGÜN, Samet Tolga ESEN

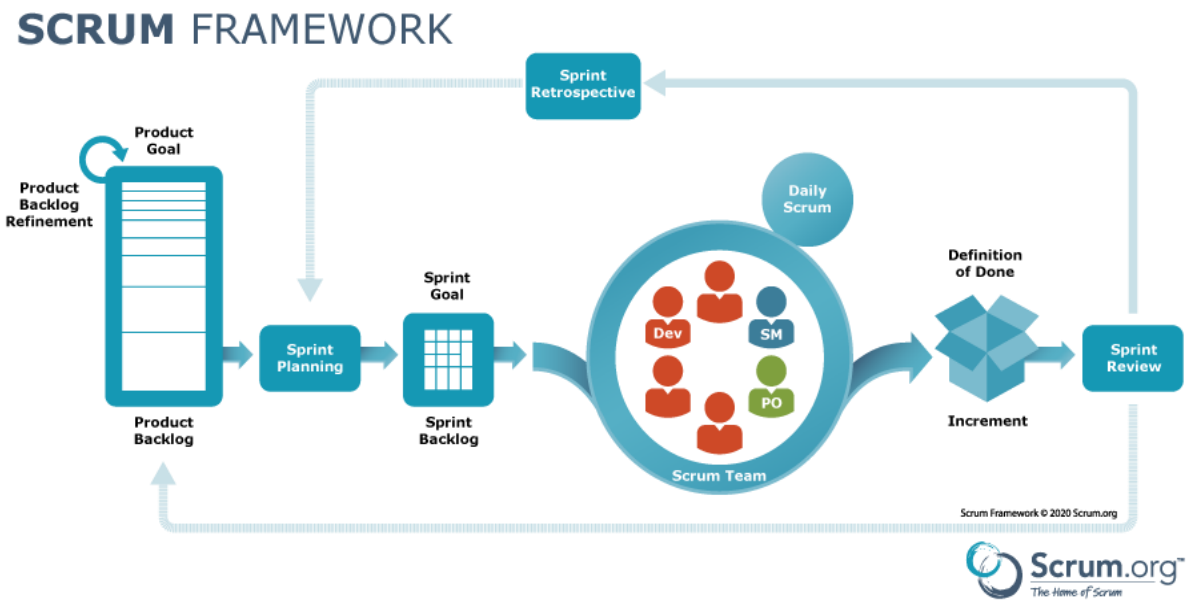
#	NECESSARY NEEDS FROM THE ORGANIZATIONAL PROCESS
1	**Efficient Prescription Management:** Given the volume of prescriptions and the need for accurate recording, efficient prescription management is crucial to streamline the process and ensure patient safety.
2	**Access Control and Authorization:** Implementing strict access controls and authorization mechanisms is necessary to maintain data security and ensure that only authorized personnel can prescribe and monitor medication.
3	**Monitoring and Audit:** Implementing monitoring and audit mechanisms is necessary to detect and address any irregularities or overprescribing practices, ensuring compliance with regulations and standards
4	**Integration with Electronic Health Systems:** Integration with existing electronic health systems is essential for keeping health records up-to-date and facilitating seamless data exchange between healthcare providers.
5	**Medication Reminder System:** Providing a medication reminder system is necessary to support medication adherence and help patients stay on track with their treatment plans.
6	**Live Support System:** Offering a live support system, including both artificial intelligence and human assistance, is necessary to provide timely and accurate assistance to users seeking information or help with their medications.
7	**Educational Resources:** Providing educational resources such as videos and animations is necessary to empower users with knowledge about their medications and promote safe and effective usage.
8	**QR Code Integration:** Integrating QR codes with prescriptions is necessary to enable quick access to medication details and prevent misuse or errors.

9	**Security and Compliance:** Ensuring data security and compliance with healthcare regulations are necessary to protect patient information and maintain trust in the system.
10	**Customer Support:** Providing reliable customer support services is necessary to address user concerns and ensure a positive user experience
#	**UNNECESSARY NEEDS FROM THE ORGANIZATIONAL PROCESS**
1	**Complexity in User Interface:** Introducing unnecessary complexity in the user interface may hinder usability and user adoption, undermining the effectiveness of the system.
2	**Redundant Features:** Including redundant features that do not add significant value to users may clutter the system and detract from its core functionality.
3	**Excessive Monitoring:** Implementing excessive monitoring without clear justification may raise privacy concerns and impact user trust in the system.
4	**Overuse of Automation:** While automation can improve efficiency, over-reliance on automation without human oversight may lead to errors or misunderstandings, particularly in sensitive healthcare contexts.
5	**Lack of Training and Support for Healthcare Providers:** Neglecting to provide adequate training and support for healthcare providers using the system may hinder adoption and effectiveness.
6	**Ignoring User Feedback:** Disregarding user feedback and preferences during the development process may result in a system that does not meet the needs or expectations of its users
7	**Overemphasis on Sales and Marketing:** While providing sales and marketing analyses to pharmaceutical companies can be beneficial, prioritizing revenue

	generation over patient safety and healthcare outcomes should be avoided.

SOFTWARE PROCESS NAME: Agile
SOFTWARE PROCESS DESCRIPTION:
The stakeholder will be consulted to determine the requirements for the initial product backlog. Product owner and stakeholders may not anticipate all requirements. After the sprint backlog is created, the targets in this backlog will be completed within a certain period of time. Begin development work on the selected backlog items, focusing on delivering working increments of functionality within each sprint. Regularly review progress and adapt plans as needed based on feedback and changing requirements. At the end of each sprint, hold a sprint review meeting to demonstrate the completed work to stakeholders and gather feedback. Use this feedback to validate assumptions, make adjustments, and refine priorities for future sprints. Conduct a sprint retrospective meeting to reflect on what went well during the sprint, what could be improved, and any actionable insights or lessons learned. Use this information to make continuous improvements to the development process. Seek regular feedback from users, stakeholders, and domain experts throughout the development process to validate assumptions, identify opportunities for improvement, and ensure that the product meets the needs of its intended users.
SOFTWARE PROCESS MODEL: Agile

SE 216 – SOFTWARE PROJECT MANAGEMENT
SOFTWARE PROCESS MODEL DOCUMENT



REASONS TO CHOOSE THIS MODEL:

Easy to Adapt: As the pharmaceutical and healthcare sectors are unpredictable, new requirements may arise. Additionally, new diseases, medicines, and laws may necessitate changes in practice, requiring us to adapt.

Manageable: The system has a high number of requirements and stakeholders, which increases the workload. However, dividing the workload into smaller parts and constantly examining and reviewing with the development team it can make it more manageable.

Visible: As progress is visible to both developers and stakeholders, they will better understand how the system works, preventing possible misunderstandings. Additionally, increased workload visibility allows for better workload management by dividing it into smaller parts.

Testable: Given the unpredictable nature of the health and pharmaceutical sector, collaborating with stakeholders can provide additional information on extreme cases. This information can help prepare for testing scenarios that developers may not have considered.

For better product: By cooperating with stakeholders and adapting as necessary, we can create a more reliable, useful, and complete application for users without relying on excessive documentation and negotiation.