Soft**Req**

Soft**Req**

Soft**Req**

****

**Integrated** Soft**ware Requirements Management Tool**

(Development)

| **Name** | **Roll Number** | **CGPA** | **Email** |
| --- | --- | --- | --- |
| Ayesha Rauf | i181594 | 2.82 | i181594@nu.edu.pk |
| Muhid Abid | i180560 | 2.65 | i180560@nu.edu.pk |
| Hassan Riaz | i180580 | 2.14 | i180580@nu.edu.pk |
| Dr. Khubaib Amjad  khubaib.amjad@nu.edu.pk | | | |



**Table of Contents**

[Problem Statement](#_a587d9rzaoki) **3**

[Motivation](#_xyb307hve079) **4**

[Stakeholders](#_99xo4yqjy87h) **5**

[Goals](#_q405odbf01cn) **6**

[Expected Outcome(s)/ Impact of the software](#_uorydmhym7g0) **7**

[High-level features](#_fphrj5pu1zx1) **8**

[Timeline](#_a8bjy3w2hjos) **10**

[Work Division](#_q5pbnvwaiti5) **11**

[FYP-1](#_6rzp4mpv7905) **11**

[FYP-2](#_79yk6pr4eyfg) **11**

[Expected tools/technologies/skills](#_nm64fl1atjwn) **12**

[References](#_6o1euahbxop9) **13**

# Problem Statement

The main goal of any software development team is to keep costs and time spent to a minimum, as well as be able to satisfy client needs. Being able to reduce revisions and redesigns during the development process as much as possible is desirable by all teams working on any sort of project.

There do exist multiple tools for requirements management. Some tools focus on analyzing and validating requirements, while others focus on managing requirements and requirement-generated tasks.

A comparison of existing market tools and their functionality in terms of requirements is given below:

| **Software** | **Collaboration** | **Requirements Classification** | **Requirements Validation** | **Requirements Management** |
| --- | --- | --- | --- | --- |
| Jama | **✔️** |  |  | **✔️** |
| IBM DOORS | **✔️** |  |  | **✔️** |
| QVScribe |  |  | **✔️** |  |
| Jira | **✔️** |  |  | **✔️** |
| SoftReq | ✔️ | ✔️ | ✔️ | ✔️ |
| (“Jama Connect vs Jira Software”), (“QVscribe by QRA Corp - Analyze Requirements Documents in Seconds”), (“Overview of DOORS”) | | | | |

SoftReq will be combining these features and more into a single integrated platform, as well as allowing real-time collaboration and requirement management between team members, for an easier, more intuitive software development process.

# Motivation

In today’s digital age, software companies are increasingly being tasked with building more and more complex products and systems. These larger systems being requested often demand carrying out extensive research, writing huge amounts of codes, even incorporating additional software. A typical software development project can be divided into more than a hundred small tasks to be completed.

In order to maintain a sense of order for any project, it’s critical to start off with a proper Requirements Management Plan. One of the stages of defining a Requirements Management Plan is to define, record, and analyze requirements. These requirements refer to what the proposed system should be able to do/achieve, the services it provides, any constraints on its operation, among other details.

The quality of these initial requirements eventually affects whether the project is successful or not, since the developers will be using these as a baseline for development. Failure to define software requirements properly upfront before development commences can lead to project scope confusion, unfulfilled client needs, project delays, increased costs as well as ambiguous standards of task completion.

To initially write these requirements, developers/business analysts have to have repeated meetings with the client and often multiple versions of the requirements are made in each meeting by multiple developers/business analysts.

We aim to ease the requirements specification process with minimalistic management yet comprehensive, real-time analysis in order to get the most complete, consistent, verifiable, independent, and valuable requirements.

# Stakeholders

| **Name** | **Represents** | **Type of Stakeholder** |
| --- | --- | --- |
| Business Analyst | The stakeholder responsible for gathering and organizing requirements from a Client before handing them off to the Project Manager. | User |
| Client | The stakeholder from whom requirements are to be elicited. | Non-User |
| Project Manager | The stakeholder who needs requirements, so development can commence. | Non-User |
| Software Developer | The stakeholder who develops a software from the requirements | Non-User |
| Software Tester | The stakeholder who tests if requirements are met by the software made by the Software Developer | Non-User |

# Goals

1. Ensure *Completeness*, *Traceability,* *Correctness,* and *Verifiability* of user stories
2. Identification of contradictions and similarities between user stories
3. Analyze user-written software requirements at run-time for ambiguities and validation errors.
4. Provide an intuitive, web-based user story editor
5. Help developers write requirements specifications in a uniform format
6. Classify each user story into
   1. Functional requirements,
   2. Quality concerns (e.g. reliability, usability),
   3. Constraints (e.g. physical, process, implementation)

# Expected Outcome(s)/ Impact of the software

This application would help bridge the gap between clients and project managers by automating the role of the business analyst. Currently, a business analyst is responsible for communicating and eliciting software requirements from the client, and then manually removing ambiguities and rewriting these requirements in a way that they become helpful to the developer.

By implementing this aspect of the work of a business analyst as a collaborative editor where both client and project manager can communicate and clarify in real-time, a business analyst will find this reduces the time spent on trying to improve user requirements, hence saving overall time in the project.

We hope that this tool will be a game-changer for startups and small to medium-sized software houses, where the team size is smaller and emphasis is on avoiding redesign as much as possible.

SoftReq will be able to cater to their needs perfectly with the following benefits:

* Minimizing time spent on validating client requirements.
* An integrated work environment with all aspects of requirement management present.
* Easier tracking of all project modules, their corresponding requirements, and further corresponding tasks.
* Helping SMEs avoid the hassle of obtaining commercial licenses for market-known requirements management software, when they only require a few key features.

# High-level features

The main deliverable from our project is a web application that assists Software developer teams in documenting, analyzing, and tracking changes in requirements in the form of user stories for any project.

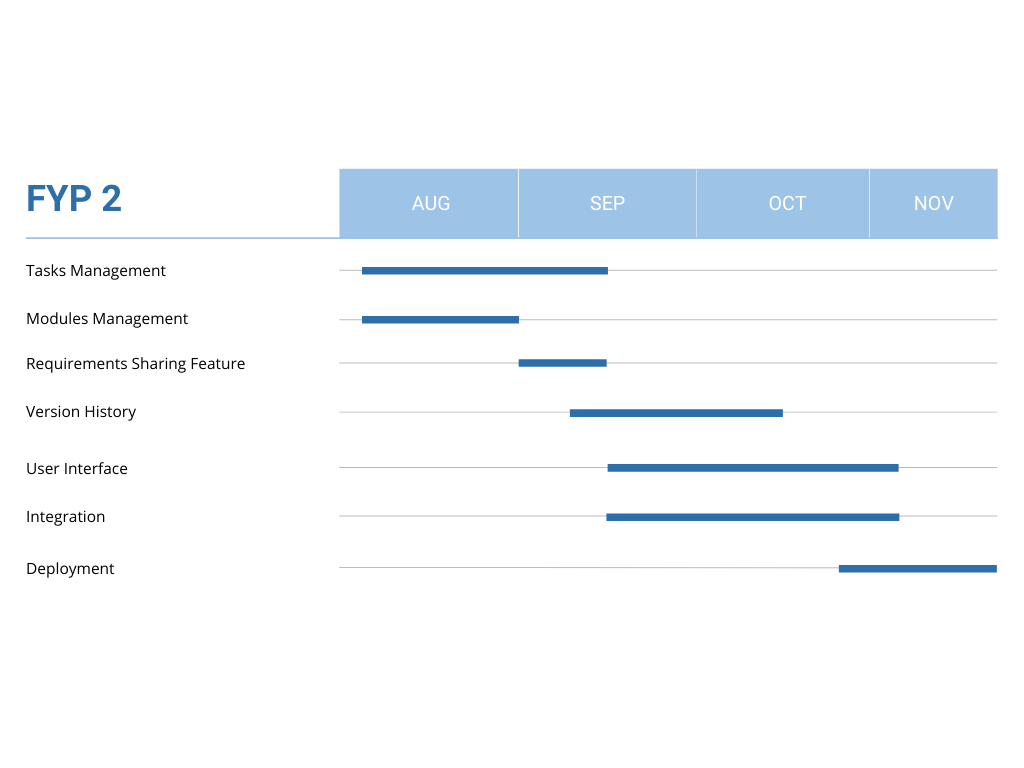
A client will be communicating their system requirements with a development team, and the team will be able to document these requirements in the Requirements Editor directly. Our tool will validate all user stories in run time. The team members can implement changes to the requirement right there and then to ensure they fulfill the characteristics of a good requirement.

While the system requirements are being collected, the Requirements Classifier will be automatically placing each finished requirement into appropriate types: functional, quality concern, or constraint. This will be useful to the development team in the organization of the requirements.

| **High-Level Feature** | **Description** |
| --- | --- |
| Requirement Editor | Users will be able to write software requirements directly in a built-in text editor. |
| Requirement Validator | The editor will validate requirements in real-time as the user is writing them out. |
| Requirement Classifier | Users will be able to see requirements classified - functional, constraints, or quality. |
| Requirements Sharing | Non-users can be given access to a User’s stored requirements and will be able to view and edit these, according to permissions given. |
| Requirements Version History | Users will be able to view previous versions of requirements and may revert a requirement back to its previous version. |
| Project Module Management | Users will be able to create project modules, and place validated requirements into appropriate modules. |
| Requirements Tasks Management | Users will be able to assign corresponding tasks to each requirement |
| Temporary Login | Users and Non-users will be able to use the Requirement Editor and Requirement Validation feature without logging into an account. |

# Timeline





# Work Division

## FYP-1

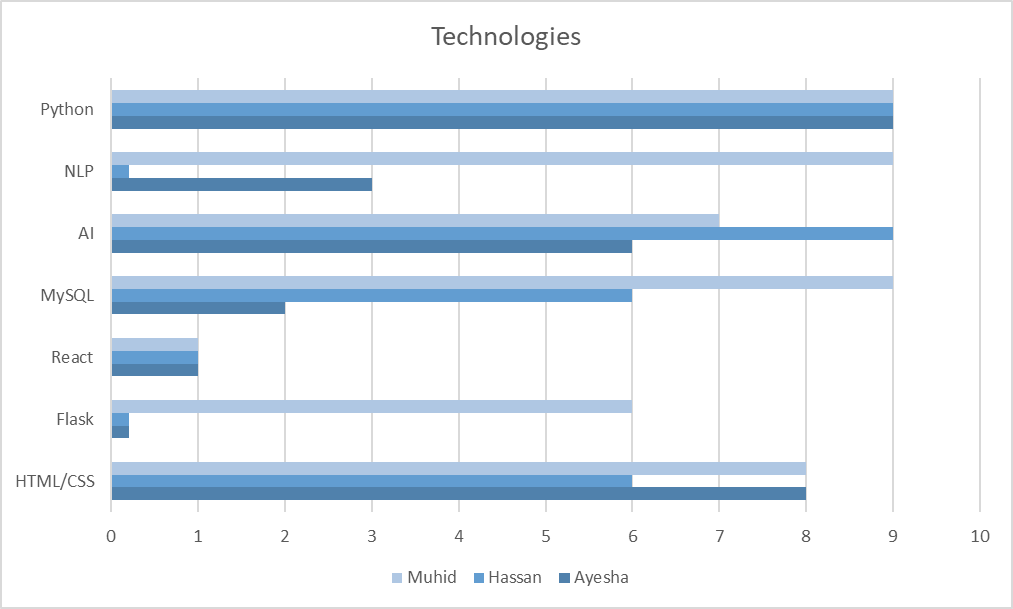
| **Task** | **Person** |
| --- | --- |
| Requirement Validation (**Ambiguity detection**) | Hassan + Ayesha |
| Requirement Validation  (**Completeness (who, why, what?)** ) | Hassan + Muhid |
| Requirement Validation **(Quantifiability detection)** | Hassan |
| Requirement Validation (**Similarity**) | Hassan + Muhid |
| Requirement Validation **(Contradiction detection)** | Ayesha + Muhid |
| Requirement Classification (**functional, constraint, quality detection - ML**) | Muhid + Ayesha + Hassan |
| Requirements Editor Backend & Integration | Muhid + Hassan + Ayesha |

## FYP-2

<To be continued>

# 

# Expected tools/technologies/skills



# References

*IEEE Recommended Practice for Software Requirements Specifications*. (2010, February 23). Computer Science and Engineering. Retrieved March 3, 2022, from http://www.cse.msu.edu/~cse870/IEEEXplore-SRS-template.pdf

*QVscribe by QRA Corp - Analyze Requirements Documents in Seconds*. (n.d.). QRA Corp. Retrieved March 3, 2022, from https://qracorp.com/qvscribe/

Requirements Management. (2015). In *Requirements Engineering Fundamentals: A Study Guide for the Certified Professional for Requirements Engineering Exam - Foundation Level - IREB Compliant*. Rocky Nook.

*Jama Connect vs Jira Software*. (n.d.). TrustRadius. Retrieved March 2, 2022, from https://www.trustradius.com/compare-products/jama-connect-vs-jira-software

Jones, C. (n.d.). *Leveraging Natural Language Processing for Requirements Analysis*. QRA Corp. Retrieved March 3, 2022, from https://qracorp.com/nlp-requirements-analysis/

*Overview of DOORS*. (n.d.). IBM. Retrieved March 2, 2022, from https://www.ibm.com/docs/en/ermd/9.7.0?topic=overview-doors

*Your Guide to Latent Dirichlet Allocation | by Lettier*. (2018, February 23). Medium. Retrieved March 3, 2022, from https://medium.com/@lettier/how-does-lda-work-ill-explain-using-emoji-108abf40fa7d

# 