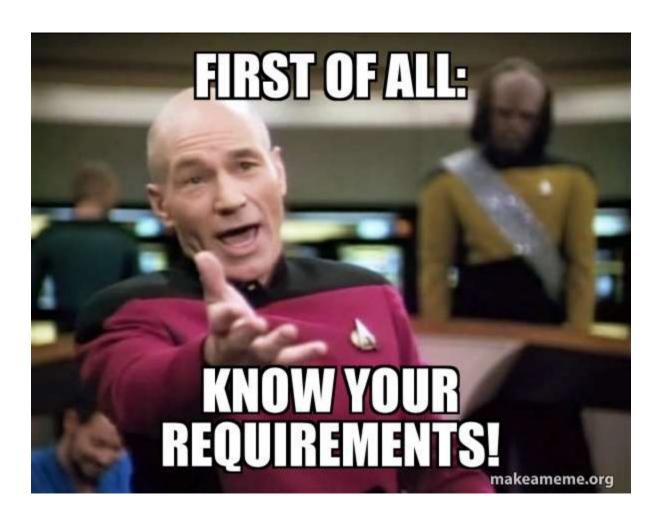
```
Understanding Software
Requirements Specification (SRS)
[S M Asiful Islam Saky]
```



Customer

Programmer













The feature does not work correctly

WHOA

This is

S M Asiful Islam Saky

Mentor, STEAM AIU CLUB









Agenda

- 1. What is SRS?
- 2. Purpose of SRS
- 3. Role of SRS
- 4. SRS Format
- 5. Readers of SRS
- 6. Characteristics of SRS
- 7. Benefits of SRS
- 8. Case Study
- 9.Q&A

What is SRS?

```
SRS is a guide for developers, telling them what to
build for a system.
It explains in detail how the system should behave and
what users expect from it.
SRS includes both what users need (user requirements)
and how the system should work (system requirements).
It describes everything the software will do and how
well it should perform.
```

```
Purpose of 'SRS' {
           The SRS document tells exactly what
     01
           the software will do.
               It helps to understand all the things
         02
               needed to make the software and to plan
                how it will look and work.
                     It also lets the customer give their
             03
                     thoughts and opinions on the software.
```

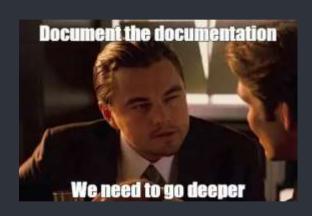
Role of 'SRS'

```
< /1 > {
         < Software Requirements Specification (SRS)
         needs to accurately describe all the things
        the software must do.
< /2 > {
         < A requirement can be needed because of the
       job the software needs to do or because of a
         unique part of the project. >
```



[SRS Format]

< Detailed Description of
writing an SRS; Template >





Characteristics of 'SRS' {

Correct

Every requirement given in SRS is a requirement of the software.

Complete

Includes all functional, performance, design, external interface requirements; definition of the response of the software to all inputs.

Unambiguous

Every requirement has exactly one interpretation.

Consistent

Internal consistency.

Ranked importance

Essential vs. desirable

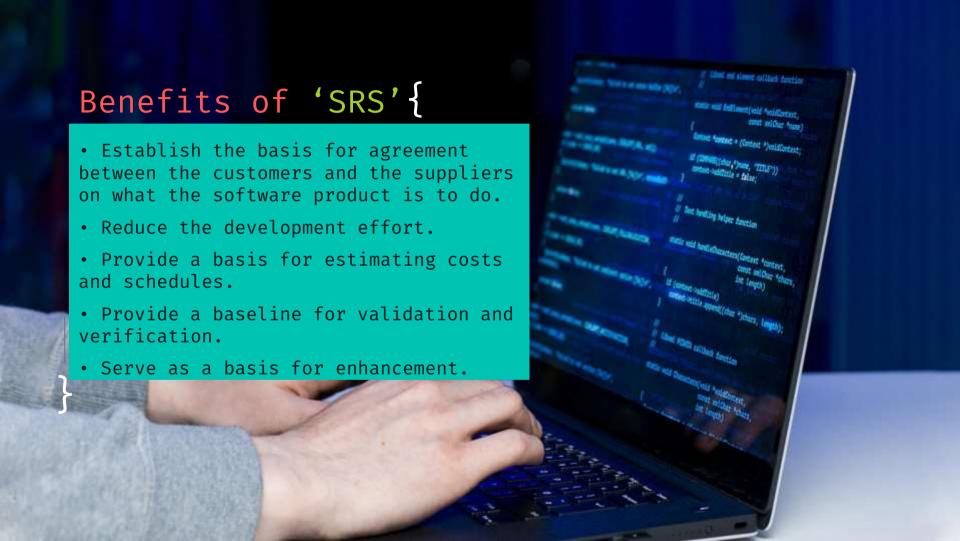
1 2 3 4 5	Title Page	Project title Version number Date Author(s) Contact information
6 7 8 9	History	Revision History Review History Approval History
10 11 12 13	Table of Contents	List of sections and subsections with page numbers

1 2 3 4	Introduction	Purpose of the document Scope of the software system Definitions, acronyms, and abbreviations
5 6 7 8 9	Overall Description	Product perspective User classes and characteristics Operating environment Design and implementation constraints Assumptions and dependencies
10 11 12 13 14	Functual Requirements	Identify and detail the product's design specifications, graphical specifications, requirements for operating systems, and any limitations or constraints.

1 2 3 4 5	External Interface Requirements	User interfaces Hardware interfaces Software interfaces Communication interfaces
6 7 8 9 10 11 12 13 14	Non-functional Requirements	Security Capacity Compatibility Reliability Scalability Maintainability Usability Other non-functional requirements

1 2 3 4	System Feature 1 System Feature 2 and so on
5 6 7 8	System Models Use case diagram State diagram
9 10 11 12 13	References and supporting documents

```
Readers of SRS {
      Business Stakeholders
      Project Managers
      Developers
      Designers
      Testers
      Regulatory Compliance Officers
      Documentation Teams
```



SRS Acceptance {

status= "accepted";

< SRS acceptance marks a pivotal
agreement point, ensuring alignment among
stakeholders and setting the foundation
for successful software development. >





https://github.com/sakysemicolon

```
SEMICOLON;
Any
Question? {
    Ask=
    "asiful.saky@student.aiu.edu.my"
    Follow =
    "www.instagram.com/saky.aiu"
    Connect =
    "www.linkedin.com/in/sakysemicol
    on"
```

