

RIS System Requirements

Functional Requirements

Organized by user type.

Referral Doctor

- Search for patient in database
- Create new patient
- Create new order for patient
- Review submitted orders
- Update individual user info

Receptionist

- Review orders
- Schedule appointments
- Check-in patients
- Review appointments
- Update individual user info

Technician

- Review orders
- Upload (imaging) files to order
- Mark order ready for diagnostic report
- Update individual user info

Radiologist

- Review orders
- Download (imaging) files from order
- Submit diagnostic report
- Update individual user info
-

Admin

- Have all privileges and abilities of all other UserRoles
- Create/modify Users and UserRoles
- Create/modify all system tables
- System tables include:
 - Users
 - UserRoles
 - Patient
 - PatientAlerts
 - Modalities
 - Orders
 - Appointments
 - FileUpload
 - DiagnosticReports

Nonfunctional Requirements

Product Requirements

- Ability to complete system process seamless from start to finish by preventing form errors from affecting other system processes
- Ability to access to file uploads remotely
- Ability to host files remotely
- User permission levels (Referral MD, Receptionist, Technician, Radiologist, Admin)
- Ability to change system options and modalities depending on radiology office
- Support access from mobile and other non-desktop devices

Organizational Requirements

- System users should be familiar with the RIS and refer to the user manual before using product
- System must be developed using Java as the primary programming language

External Requirements

- All users of a medical information system should adhere to proper information confidentiality guidelines and laws
- External users such as referral doctors should not have access to medical records of patients other than those they have placed orders for
- Patient information should be protected in such a way that data leaks will not pose a serious risk to the identity of individuals
 - This includes the encryption of file uploads upon submission