Engineering RFC - Template

# **Authors**

* Suyash Kumar - [suyash@suyashkumar.com](mailto:suyash@suyashkumar.com)

# **Reviewers**

* Suyash Kumar - [suyash@suyashkumar.com](mailto:suyash@suyashkumar.com) - PENDING

# **Table of Contents**

[Authors](#_qqarhx4k57oi)

[Reviewers](#_590osg2th4ii)

[Table of Contents](#_vdnen7g1xow9)

[Abstract](#_gu2rsd84k0vu)

[Background](#_572uukem0bkd)

[Designs [Mostly for frontend]](#_q9v4vu1i8xpy)

[Architecture](#_oagse2b55tq9)

[Diagram](#_x978okmqhzk2)

[Endpoints](#_zc4cgsyn7ei4)

[Database](#_vnot6u41j19w)

[Infrastructure](#_koojoirun0vw)

[Security Considerations](#_rrlr94h083xn)

[Failure Modes & Mitigation](#_3e3byfu3t1ln)

[Alerting & Monitoring](https://docs.google.com/document/d/1_CWGZIxhn--o6uaCprvCYq_HpWJHKdBkE9Nzy9Vk6Hg/edit#heading=h.ry9o46f3mwnj)

# **Abstract**

Briefly describe what this service/feature does and why. You may wish to highlight certain implementation details here.

# **Background**

If there is a *Feature Narrative* or *Product Problem Statement*, link it here. Describe the motivations and goals of this product, along with any backstory that’ll help RFC readers quickly understand relevant context.

# **Designs [Mostly for frontend]**

Link/paste any designs/mocks here.

# **Architecture**

Modify this section to best convey the technical design and architecture of what you’re building. Some suggested subsections are below. Be sure to include what technologies are being used (and why if applicable).

## **Diagram**

If it makes sense, draw a lucidchart diagram depicting how this service functions and what it depends on.

## **Endpoints**

List endpoints and their interfaces here. Link to draft protobufs if possible for exact clarity on the API.

## **Database**

Be sure to clearly outline database schemas and future migration strategies if applicable.

# **Infrastructure**

1. Where will instances of this feature/service run (within Hospital networks, cloud, etc).
2. Does this feature/service have specific infrastructure requirements?

# **Security Considerations**

1. Does this service handle and/or persist PHI? If so, what precautions will be taken to safeguard this data?
2. Does this service rely on secrets? How are those secrets distributed/safeguarded?

# **Failure Modes & Mitigation**

1. What are the service dependencies?
2. What problems occur if those downstream services are malfunctioning (high latencies) or down altogether?
3. What other failure modes exist for this feature/service, and what impact do those have? How can they be mitigated?

# **Alerting & Monitoring**

1. What logging and monitoring will be in place for this feature/service
2. What alerts make sense for this service (if any) and how will those alerts be configured and set?