

Semester 2 SA Assignment

Vilas Varghese

Submitted on 28th Feb, 2022

Goal

- Online discussion forum for employees in a company to learn and share their knowledge

Key requirements of the system

- Key requirements of the system can be categorized as
 - Functional requirements
 - Non-Functional requirements – can be categorized as
 - Quality Attributes
 - Constraints

Functional Requirement

- Actors involved are
 - Employee
 - Moderators
 - Administrators
- Integrate with another internal service for authentication and authorization
- An employee can
 - login to the collaboration portal through sso
 - search and view questions.
 - post new questions.
 - answer/reply to question.
 - upvote a question/answer.
 - flag a question, answer for moderator attention.
 - earn badges for being helpful.
 - vote to close a question
 - mark a question as confidential

Functional Requirement

- A moderator can
 - Accept/Decline a question/answer as confidential so that it will be visible only to restricted group or to everyone.
 - Accept/Decline a flagged question or answer so that it is soft deleted or not.
- An administrator can
 - CRUD category for discussion
 - Add/remove an user to the moderator group.

Non Functional Requirements

- There can be enterprise specific confidential discussions. Security is of prime importance.
- Each page should be loaded in less than 2 seconds.
- Initially design for a single Business unit. Open to entire organization shortly.
- If the usage grows, there could be proprietary restrictions. Hence we may have to port/scale the solution across to other BU's.
- There can be client specific customization request during such porting. Open up discussion portal for other partners working for current client. Usability may not be a priority now.
- Application should be automation-tested to support frequent releases.

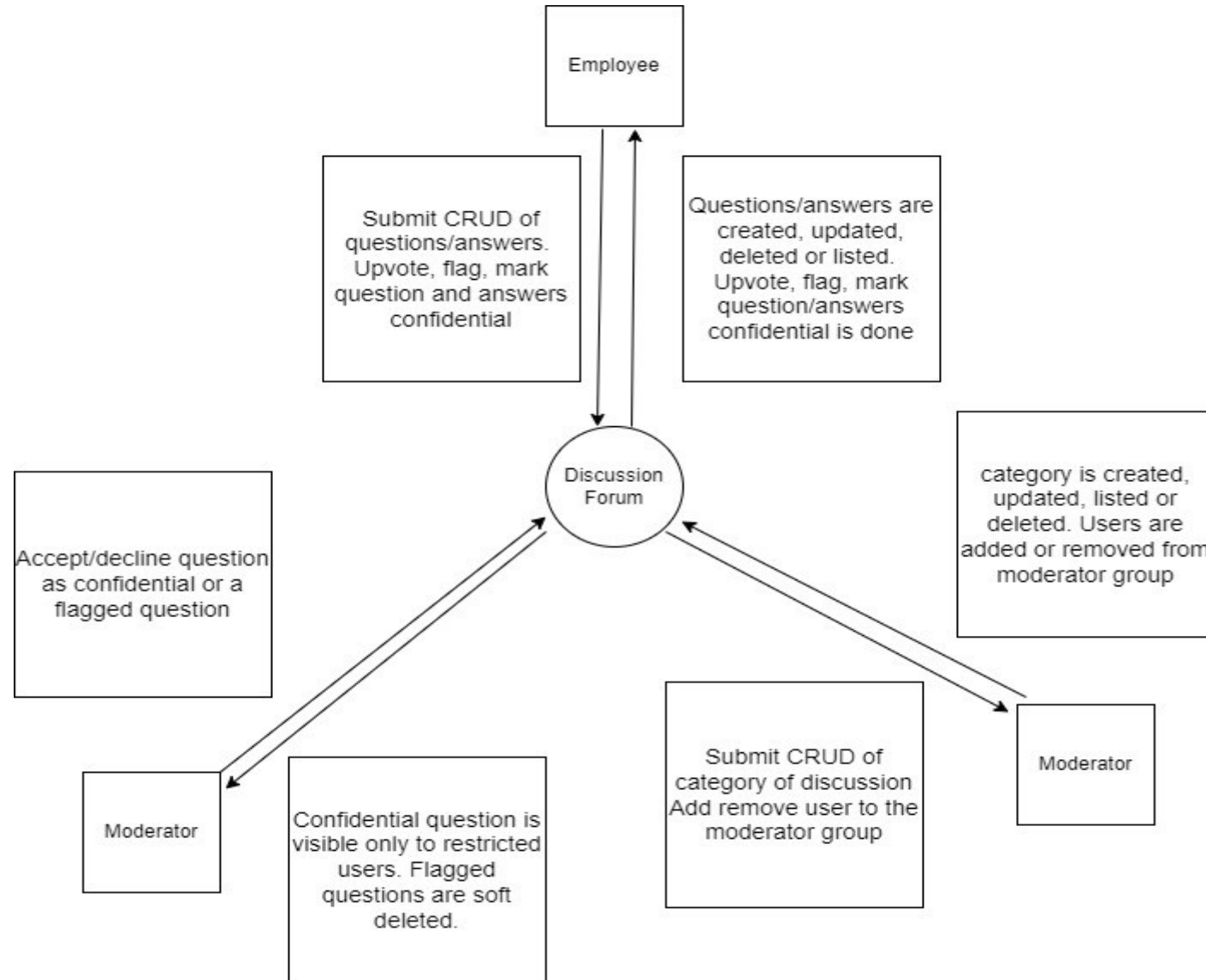
- Constraints

- MVP should be build in 6 months

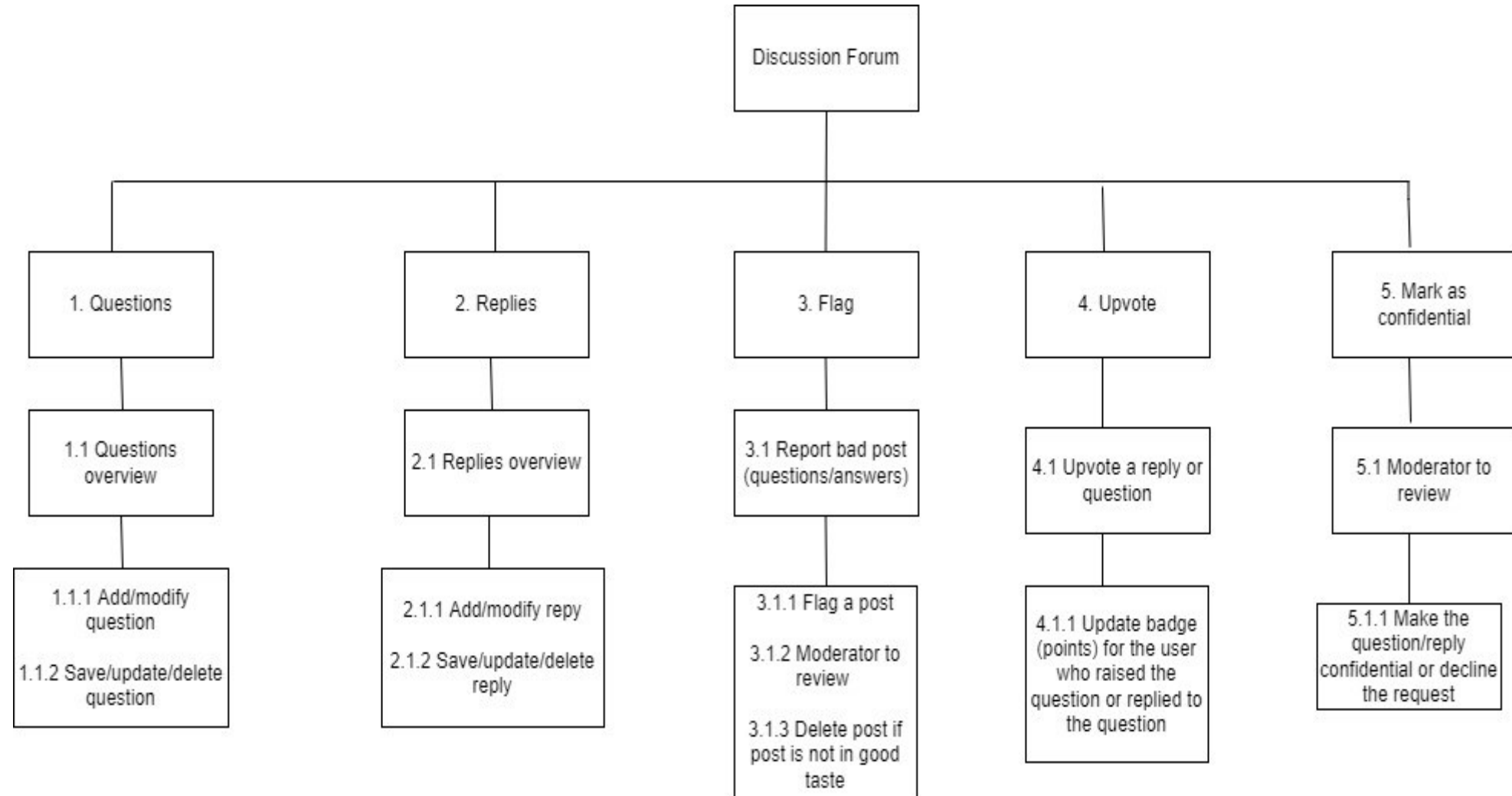
ASR

- Security is of prime importance. Authentication should happen by integrating with internal SSO. Critical data should be encrypted and presented with appropriate MFA.
- Performance: Each page should be loaded in less than 2 seconds
- Availability: Should be available to the customer - downtime is not acceptable. 100% available.
- Testability. Support automation-testing to do frequent releases.
- Modifiability. Requirements can change as we open it up for wider audience.

Context Diagram



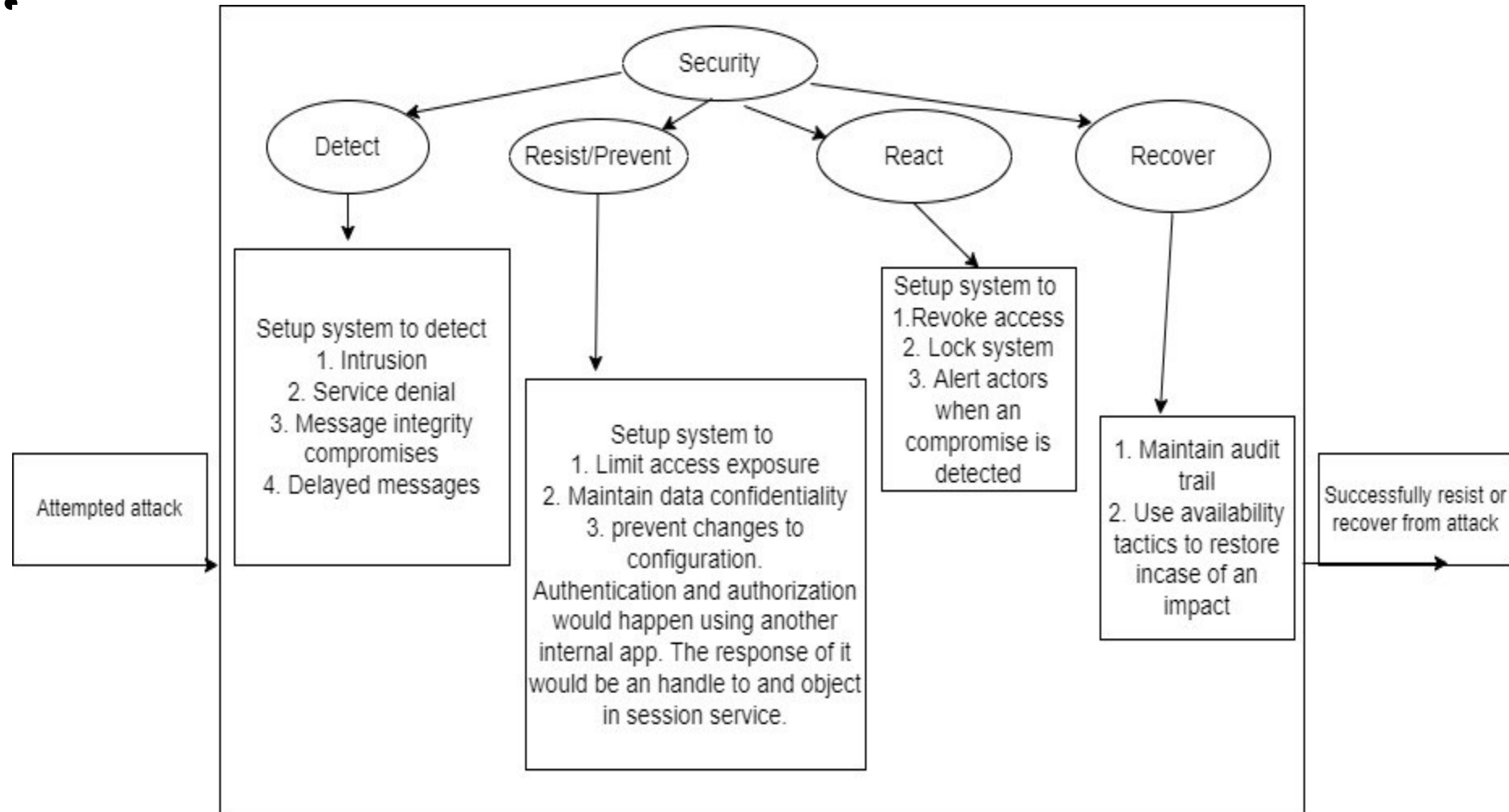
Functional Decomposition



Architectural diagram

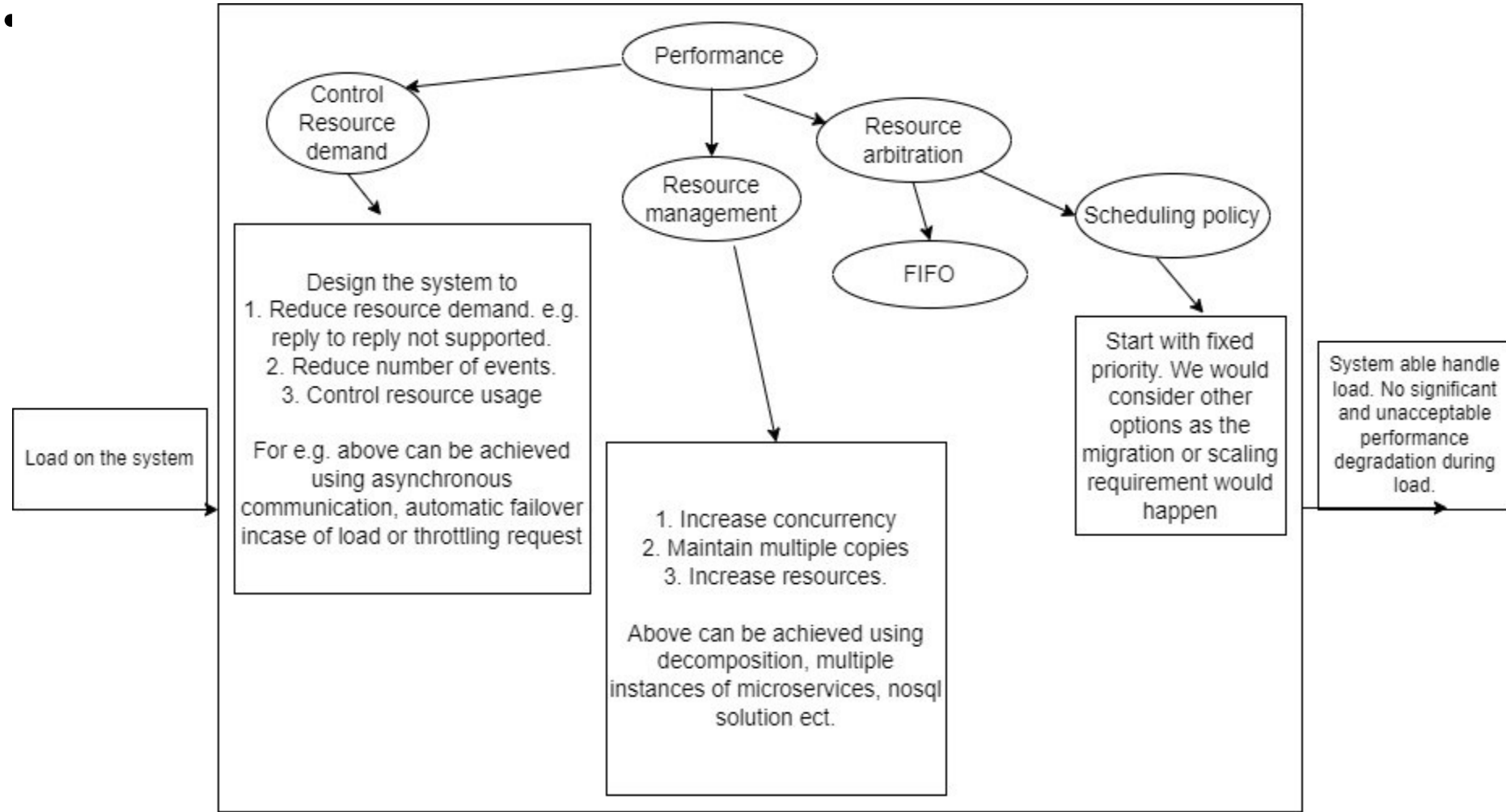
- The top 5 ASR are
 - Security
 - Performance
 - Availability
 - Testability
 - Modifiability

ASR Security



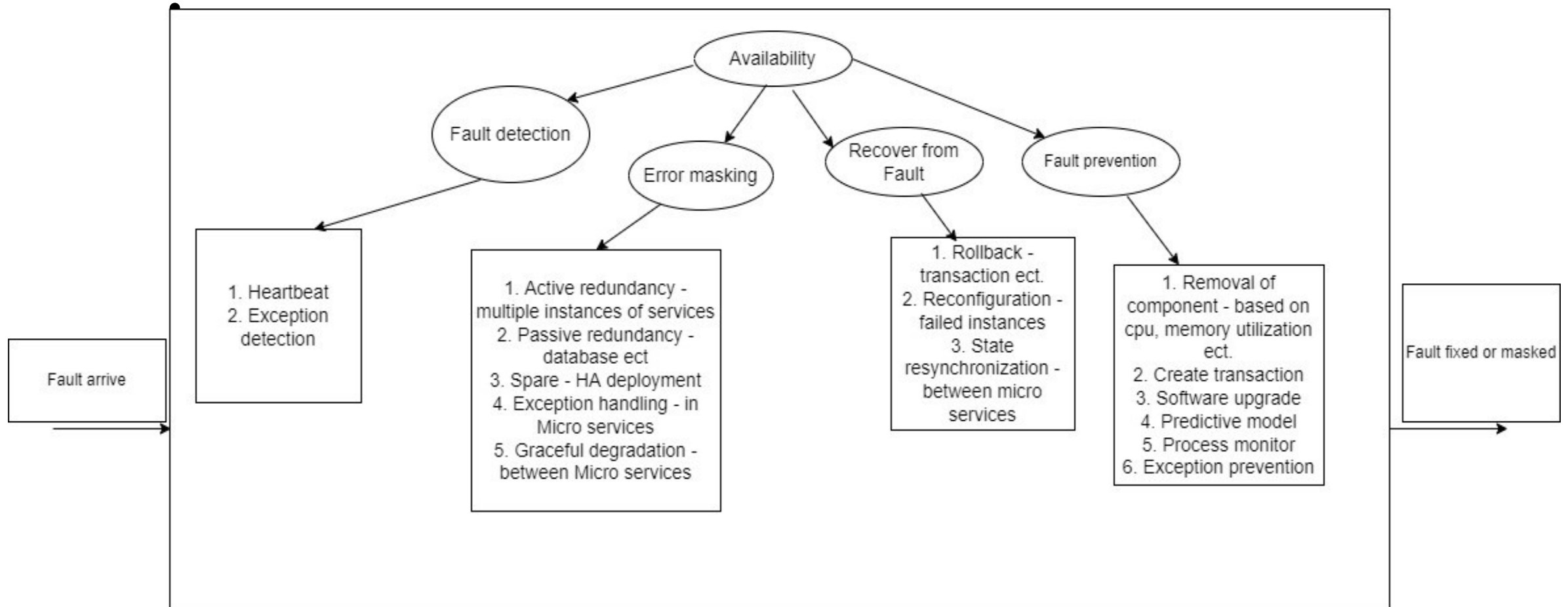
*Diagram shows only the options selected in the current architecture

ASR – Performance



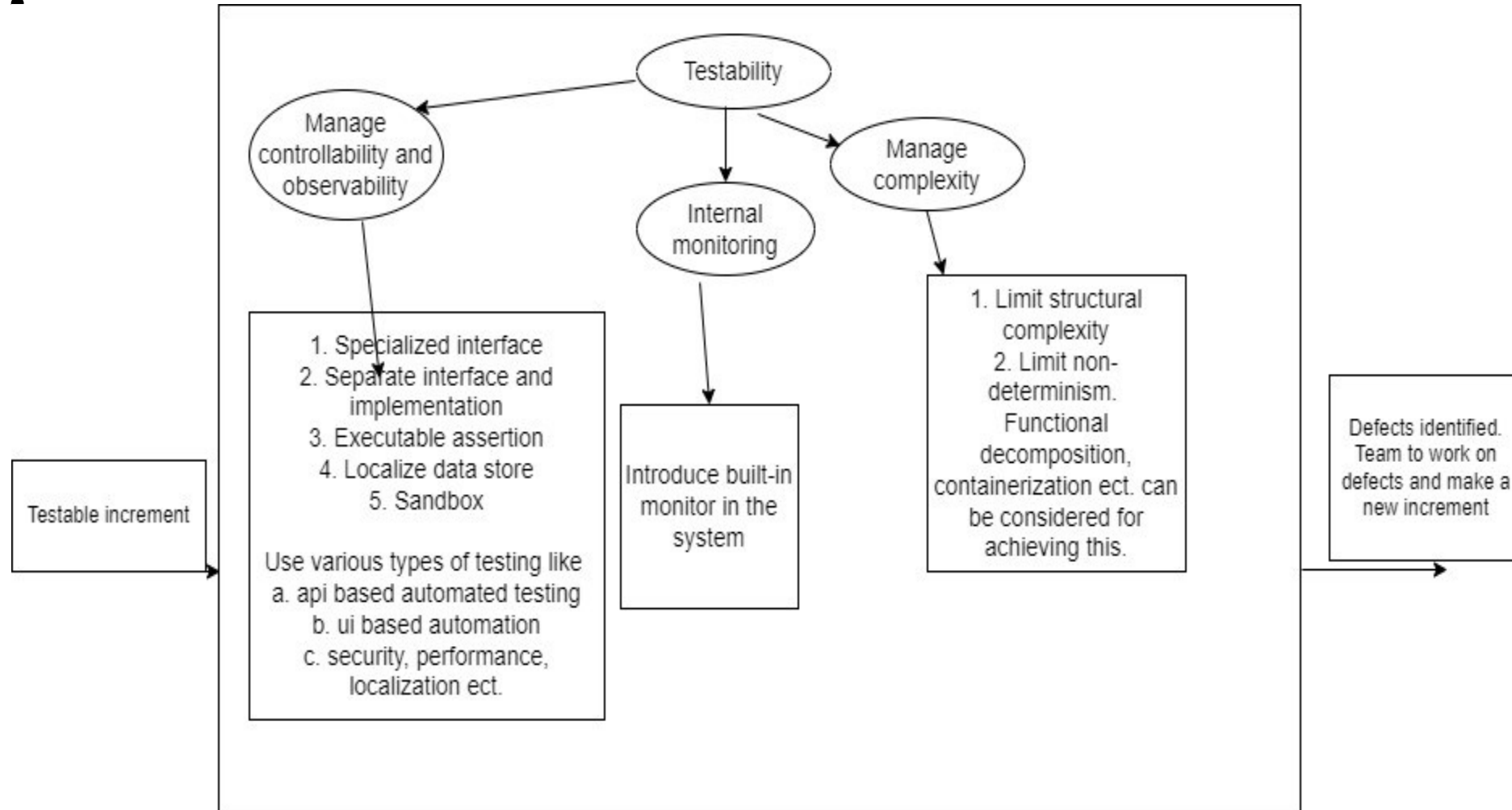
*Diagram shows only the options selected in the current architecture

ASR – Availability



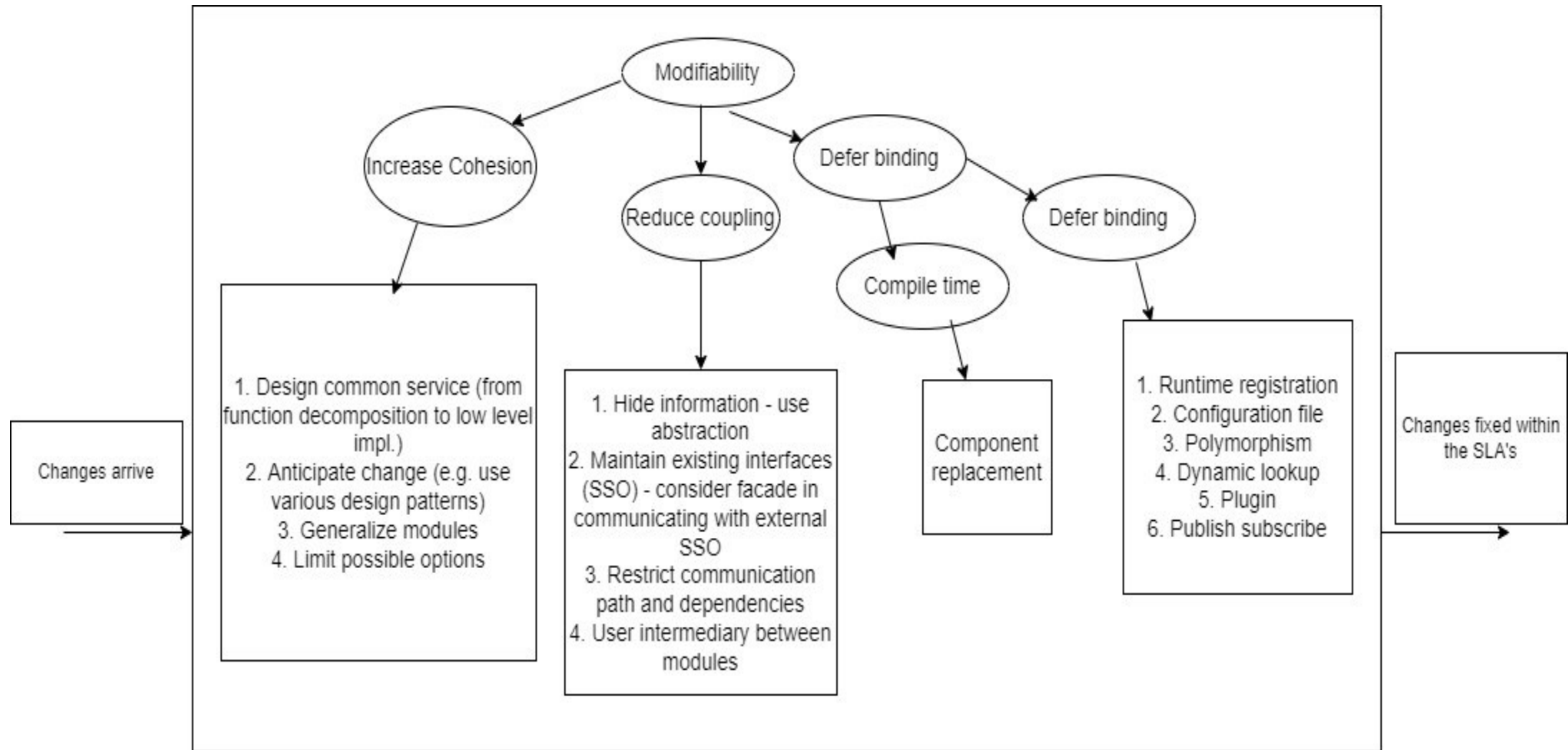
*Diagram shows only the options selected in the current architecture

ASR - Testability



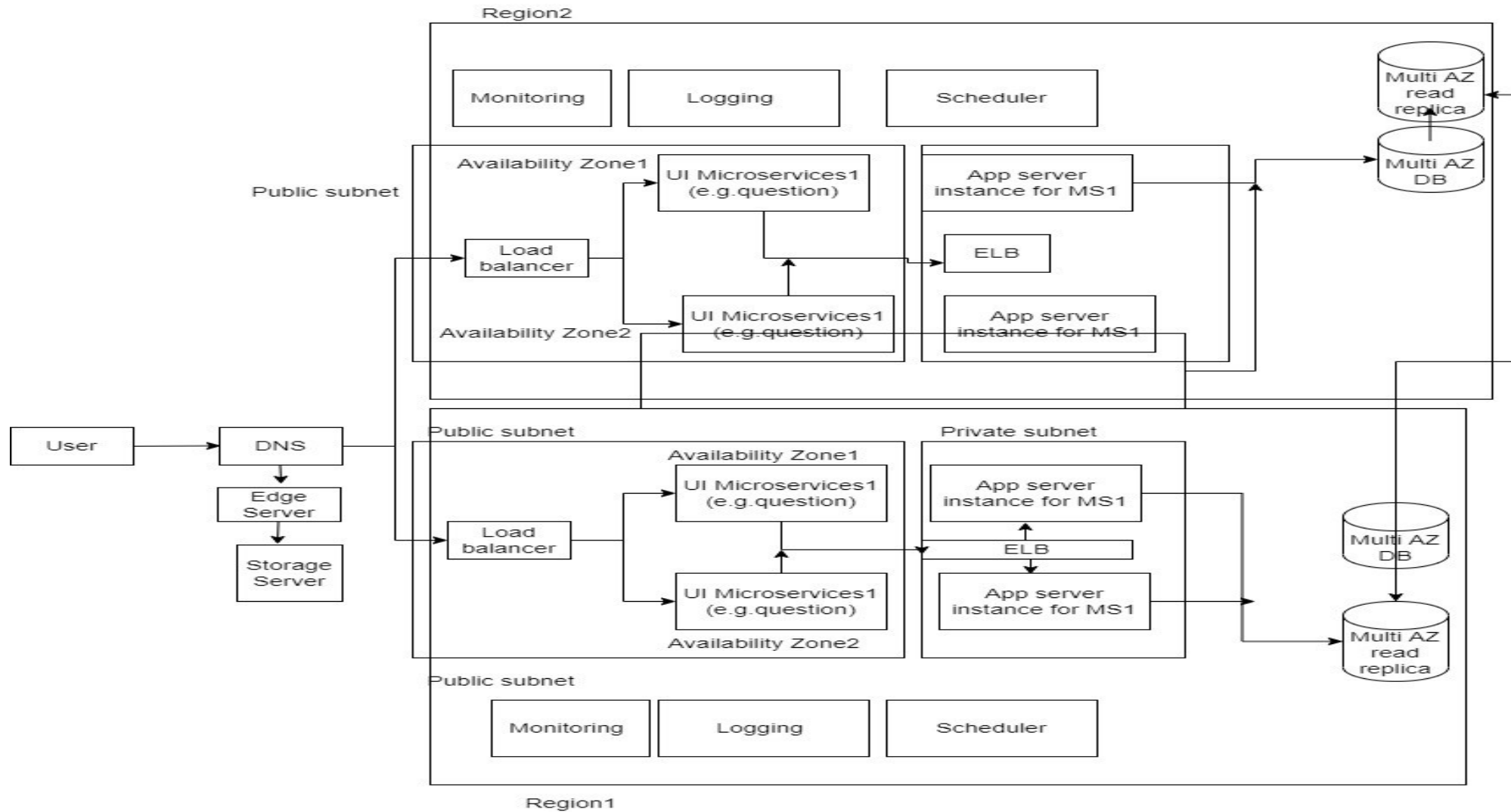
*Diagram shows only the options selected in the current architecture

ASR - Modifiability

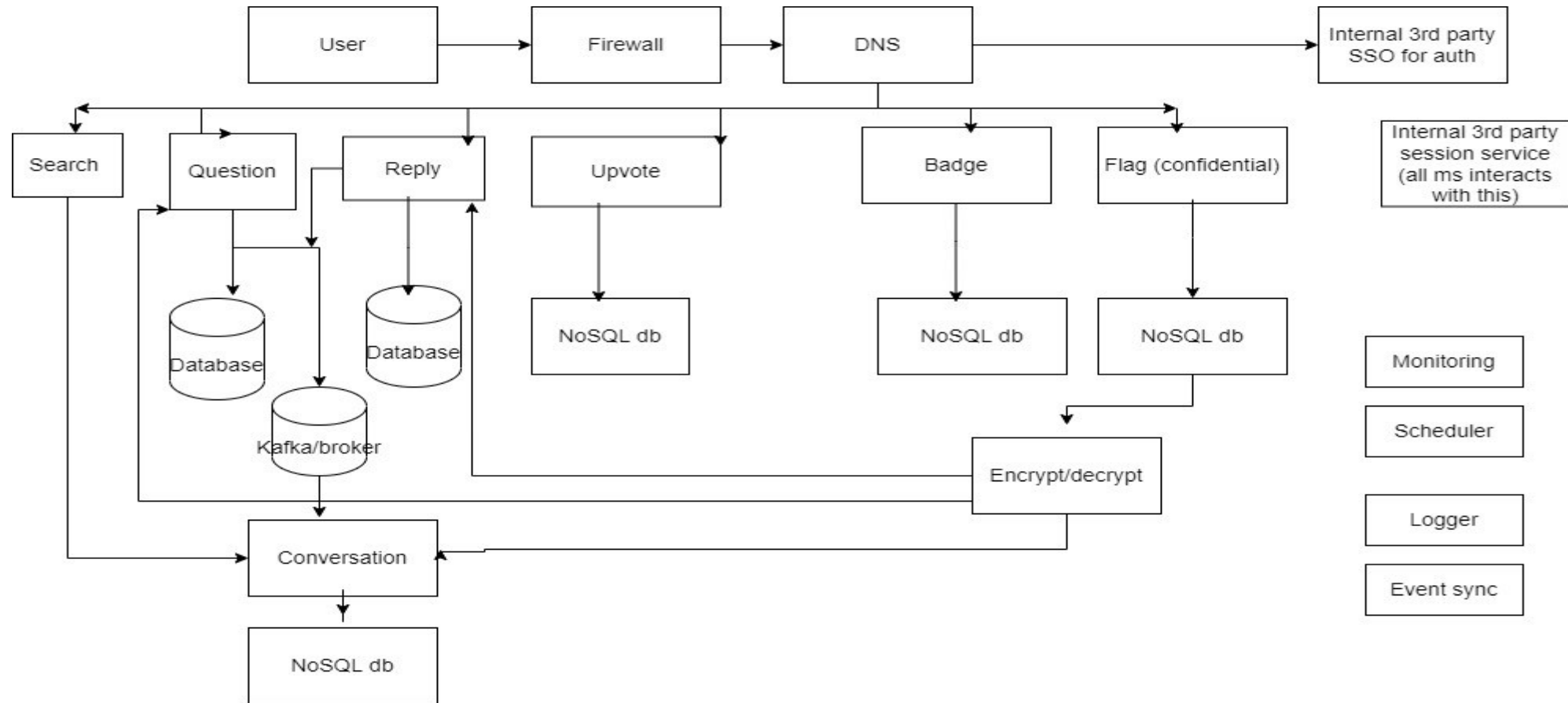


*Diagram shows only the options selected in the current architecture

Deployment Diagram



Component connector diagram



Region2

*Diagram contains only the question section, Other sections like notifications ect. We not considered in this to restrict the size.

How the system works

- We expect to scale the requirement in a short time. Hence there is an edge server which caching the static content.
- Integrate with internal application maintained by different team for SSO, authentication and authorization.
- Request from the user is routed to the right microservice using path based routing and load balanced to the appropriate instance of the microservice.
- Question and replies are maintained in relational database to ensure ACID properties. However there is a consolidated data prepared using asynchronous call to Conversation service.

How the system works cntd.

- Conversation service will be single point for searching the data.
- Flagging, badge ect are maintained in the nosql database to improve scalability.
- Flagged content is encrypted using an encryption service. The key can be maintained in a vault.

How did we meet the ASR?

- Below points are in addition to the ASR and tactics diagram.
- Security:
 - Confidential (Flagged) data is encrypted and visible only to restricted group.
 - Leverage the SSO and MFA at enterprise to drive security.
- Performance:
 - Functionalities are decomposed and deployed to scale.
 - Usage of NoSQL at appropriate levels also improves redundancy.
 - Horizontal pod auto scaling, containerization and orchestration ect. are additional considerations to meet performance demands.
- Availability:
 - Pod's are distributed across Availability zones.
 - Fault tolerance is further improved by an active failover region that should be geographically in a different far away location.
 - Further 6 level db backup ect. Should improve data redundancy and Disaster recovery.

How did we meet the ASR? Cntd.

- Testability.
 - Automate testing
 - Unit tests
 - Integration tests
 - Performance
 - Security
 - Static analysis
 - Ect.

Key Learning

- Identifying the NFR's are critical and complicated.
- Priority of ASR's could completely different between different products. Prior experience can be used to frame questions but not make assumptions.
- Analyzing multiple solutions and weighing the result takes time. Taking an agile approach to build a working sample helps.
- Beyond understanding the functional and non-functional requirement, understanding the existing (successful) solutions is an important point to be considered.
- Identifying the right stakeholders is also very important in identifying the right functional and non-functional requirements.

Thank You!!