



DevOps
Conf 2022

Проверка навыков SRE:
собеседования по
system design и
troubleshooting

Поломодов Александр

Tinkoff



Привет!

Меня зовут Александр

Руководжу управлением разработки цифровых экосистем в Tinkoff



Привет!

Меня зовут Александр

Руководжу управлением разработки цифровых экосистем в Tinkoff
Курирую наши System Design Interview и
Выступаю в качестве интервьюера по Troubleshooting



План нашей беседы

X Как выглядит процесс найма SRE



План нашей беседы

- X Как выглядит процесс найма
- X Зачем мы проверяем навыки *troubleshooting* у кандидатов



План нашей беседы

- X Как выглядит процесс найма
- X Зачем мы проверяем навыки *troubleshooting* у кандидатов
- X Чем это полезно самому кандидату



План нашей беседы

- X Как выглядит процесс найма
- X Зачем мы проверяем навыки *troubleshooting* у кандидатов
- X Чем это полезно самому кандидату
- X Как выглядит секция *troubleshooting*



План нашей беседы

- X Как выглядит процесс найма
- X Зачем мы проверяем навыки *troubleshooting* у кандидатов
- X Чем это полезно самому кандидату
- X Как выглядит секция *troubleshooting*
- X Как выглядит типовая задача



План нашей беседы

- X Как выглядит процесс найма
- X Зачем мы проверяем навыки *troubleshooting* у кандидатов
- X Чем это полезно самому кандидату
- X Как выглядит секция *troubleshooting*
- X Как выглядит типовая задача
- X Как мы оцениваем кандидатов



План нашей беседы

- X Как выглядит процесс найма
- X Зачем мы проверяем навыки *troubleshooting* у кандидатов
- X Чем это полезно самому кандидату
- X Как выглядит секция *troubleshooting*
- X Как выглядит типовая задача
- X Как мы оцениваем кандидатов
- X Как выглядит секция *system design*



План нашей беседы

- X Как выглядит процесс найма
- X Зачем мы проверяем навыки *troubleshooting* у кандидатов
- X Чем это полезно самому кандидату
- X Как выглядит секция *troubleshooting*
- X Как выглядит типовая задача
- X Как мы оцениваем кандидатов
- X Как выглядит секция *system design*
- X В чем отличия *troubleshooting* и *system design*



План нашей беседы

- X Как выглядит процесс найма
- X Зачем мы проверяем навыки *troubleshooting* у кандидатов
- X Чем это полезно самому кандидату
- X Как выглядит секция *troubleshooting*
- X Как выглядит типовая задача
- X Как мы оцениваем кандидатов
- X Как выглядит секция *system design*
- X В чем отличия *troubleshooting* и *system design*
- X Как прокачаться в *system design* и *troubleshooting*



Как выглядит процесс найма для SRE

Candidate Level



Senior+



Senior



Middle



Junior

Hiring stages

Candidate Level



Senior+



Senior



Middle



Junior

Pre-interview

Technical stages

Fit
Interview

Offer

Hiring stages

Candidate Level



Senior+



Senior



Middle



Junior

Pre-interview

Interview with
a recruiter

Technical stages

Fit
Interview

Offer

Hiring stages

Candidate Level



Senior+



Senior



Middle



Junior

Pre-interview

Interview with
a recruiter

Technical stages

System
Engineering

Fit
Interview

Offer

Hiring stages

Candidate Level



Senior+



Senior



Middle



Junior

Pre-interview

Interview with
a recruiter

Technical stages

System
Engineering

Programming

Fit
Interview

Offer

Hiring stages

Candidate Level



Senior+



Senior



Middle



Junior

Pre-interview

Interview with
a recruiter

Technical stages

System
Engineering

Programming

Troubleshooting

Fit
Interview

Offer

Hiring stages

Candidate Level



Senior+



Senior



Middle



Junior

Pre-interview

Interview with
a recruiter



Technical stages

System
Engineering



Programming



Troubleshooting



Fit
Interview



Offer



Hiring stages



Зачем мы проверяем
навыки *troubleshooting*
у кандидатов?



Зачем мы проверяем навыки troubleshooting у кандидатов

X 25+ млн клиентов

Быстрый рост количества клиентов и нагрузки



Зачем мы проверяем навыки troubleshooting у кандидатов

- X 25+ млн клиентов
- X Мультипродуктовая компания

Быстрый рост количества клиентов и нагрузки
Много отдельных бизнес-линий, платформ и сервис-линий со своими IT-подразделениями



Зачем мы проверяем навыки troubleshooting у кандидатов

- X 25+ млн клиентов
- X Мультипродуктовая компания
- X Продукты объединяются в экосистему

Быстрый рост количества клиентов и нагрузки

Много отдельных бизнес-линий, платформ и сервис-линий со своими IT-подразделениями

Большое количество разнообразных нетривиальных интеграций между продуктами



Зачем мы проверяем навыки troubleshooting у кандидатов

- | | |
|--------------------------------------|---|
| X 25+ млн клиентов | Быстрый рост количества клиентов и нагрузки |
| X Мультипродуктовая компания | Много отдельных бизнес-линий, платформ и сервис-линий со своими IT-подразделениями |
| X Продукты объединяются в экосистему | Большое количество разнообразных нетривиальных интеграций между продуктами |
| X Быстрый рост штата | Монолитные команды делятся на кросс-функциональные и автономные продуктовые команды |



Зачем мы проверяем навыки troubleshooting у кандидатов

- | | |
|---|---|
| X 25+ млн клиентов | Быстрый рост количества клиентов и нагрузки |
| X Мультипродуктовая компания | Много отдельных бизнес-линий, платформ и сервис-линий со своими IT-подразделениями |
| X Продукты объединяются в экосистему | Большое количество разнообразных нетривиальных интеграций между продуктами |
| X Быстрый рост штата | Монолитные команды делятся на кросс-функциональные и автономные продуктовые команды |
| X Децентрализация обеспечения надежности систем | Инженеры в командах должны сами уметь обеспечивать надежность своих систем |



Чем это полезно
самому кандидату



Чем это полезно самому кандидату

X Возможность получить интересный опыт в работе над инцидентом в условиях приближенных к реальности

Обычно такой опыт можно получить только в условиях работы над реальной системой, когда цена ошибки значительна. В интервью же цена ошибки не так велика и можно проверить себя почти в «бою»



Чем это полезно самому кандидату

X Возможность получить интересный опыт в работе над инцидентом в условиях приближенных к реальности

Обычно такой опыт можно получить только в условиях работы над реальной системой, когда цена ошибки значительна. В интервью же цена ошибки не так велика и можно проверить себя почти в «бою»

X В зависимости от результатов собеседования мы дадим разную обратную связь

Если собеседование пройдено

- Круто – вероятно предложим позицию Senior/Lead
- Нормально – позовем в команду, где можно прокачаться
- Так себе – порекомендуем что подтянуть



Как выглядит troubleshooting секция

0

Описание регламента
собеседования:

- Мы вместе SRE инженеры
 - Кандидат Lead, а интервьюер - джун
 - Кандидат в отпуске,
- начинается инцидент и интервьюер звонит ему и просит помочь



1

Описание существующей системы:

- Какая функциональность предоставляется
- Какие компоненты есть
- Как они связаны



0

Описание регламента собеседования:

- Мы вместе SRE инженеры
- Кандидат Lead, а интервьюер - джун
- Кандидат в отпуске, начинается инцидент и интервьюер звонит ему и просит помочь



Описание существующей системы:

- Какая функциональность предоставляется
- Какие компоненты есть
- Как они связаны



Формализация задачи –
любые уточнения по
задаче



Описание регламента
собеседования:

- Мы вместе SRE инженеры
- Кандидат Lead, а интервьюер - джун
- Кандидат в отпуске, начинается инцидент и интервьюер звонит ему и просит помочь



1

Описание существующей системы:

- Какая функциональность предоставляется
- Какие компоненты есть
- Как они связаны



2

Формализация задачи –
любые уточнения по задаче



3

Старт инцидента и
описание наблюдаемых
симптомов



0

Описание регламента
собеседования:

- Мы вместе SRE инженеры
 - Кандидат Lead, а интервьюер – джун
 - Кандидат в отпуске,
- начинается инцидент и интервьюер звонит ему и просит помочь



1

Описание существующей системы:

- Какая функциональность предоставляется
- Какие компоненты есть
- Как они связаны



0

Описание регламента собеседования:

- Мы вместе SRE инженеры
- Кандидат Lead, а интервьюер - джун
- Кандидат в отпуске, начинается инцидент и интервьюер звонит ему и просит помочь



2

Формализация задачи – любые уточнения по задаче



3

Старт инцидента и описание наблюдаемых симптомов



4

Диагностика проблем при помощи формулирования гипотез и проведения экспериментов над системой



1

Описание существующей системы:

- Какая функциональность предоставляется
- Какие компоненты есть
- Как они связаны



2

Формализация задачи – любые уточнения по задаче



3

Старт инцидента и описание наблюдаемых симптомов



0

Описание регламента собеседования:

- Мы вместе SRE инженеры
- Кандидат Lead, а интервьюер – джун
- Кандидат в отпуске, начинается инцидент и интервьюер звонит ему и просит помочь



4

Диагностика проблем при помощи формулирования гипотез и проведения экспериментов над системой



5

Проблему починили при помощи workaround решения



1

Описание существующей системы:

- Какая функциональность предоставляется
- Какие компоненты есть
- Как они связаны



0

Описание регламента собеседования:

- Мы вместе SRE инженеры
- Кандидат Lead, а интервьюер - джун
- Кандидат в отпуске, начинается инцидент и интервьюер звонит ему и просит помочь



2

Формализация задачи – любые уточнения по задаче



3

Старт инцидента и описание наблюдаемых симптомов



4

Диагностика проблем при помощи формулирования гипотез и проведения экспериментов над системой



5

Проблему починили при помощи workaround решения



6

Проблему починили и кандидат может сформулировать алгоритм для исправления



1

Описание существующей системы:

- Какая функциональность предоставляется
- Какие компоненты есть
- Как они связаны



2

Формализация задачи – любые уточнения по задаче



3

Старт инцидента и описание наблюдаемых симптомов



0

Описание регламента собеседования:

- Мы вместе SRE инженеры
- Кандидат Lead, а интервьюер – джун
- Кандидат в отпуске, начинается инцидент и интервьюер звонит ему и просит помочь



4

Диагностика проблем при помощи формулирования гипотез и проведения экспериментов над системой



7

Кандидат разобрался в причинах проблемы и может объяснить изначально наблюдаемые симптомы и почему алгоритм исправления сработал



6

Проблему починили и кандидат может сформулировать алгоритм для исправления



5

Проблему починили при помощи workaround решения



1

Описание существующей системы:

- Какая функциональность предоставляется
- Какие компоненты есть
- Как они связаны



2

Формализация задачи – любые уточнения по задаче



3

Старт инцидента и описание наблюдаемых симптомов



0

Описание регламента собеседования:

- Мы вместе SRE инженеры
- Кандидат Lead, а интервьюер – джун
- Кандидат в отпуске, начинается инцидент и интервьюер звонит ему и просит помочь



8

Кандидат рассказал как улучшить систему, чтобы

- Похожие проблемы не повторялись в будущем
- Или как минимум мы о них оперативно узнавали



4

Диагностика проблем при помощи формулирования гипотез и проведения экспериментов над системой



7

Кандидат разобрался в причинах проблемы и может объяснить изначально наблюдаемые симптомы и почему алгоритм исправления сработал



6

Проблему починили и кандидат может сформулировать алгоритм для исправления



5

Проблему починили при помощи workaround решения





Как выглядит типовая
задача



Описание регламента
собеседования



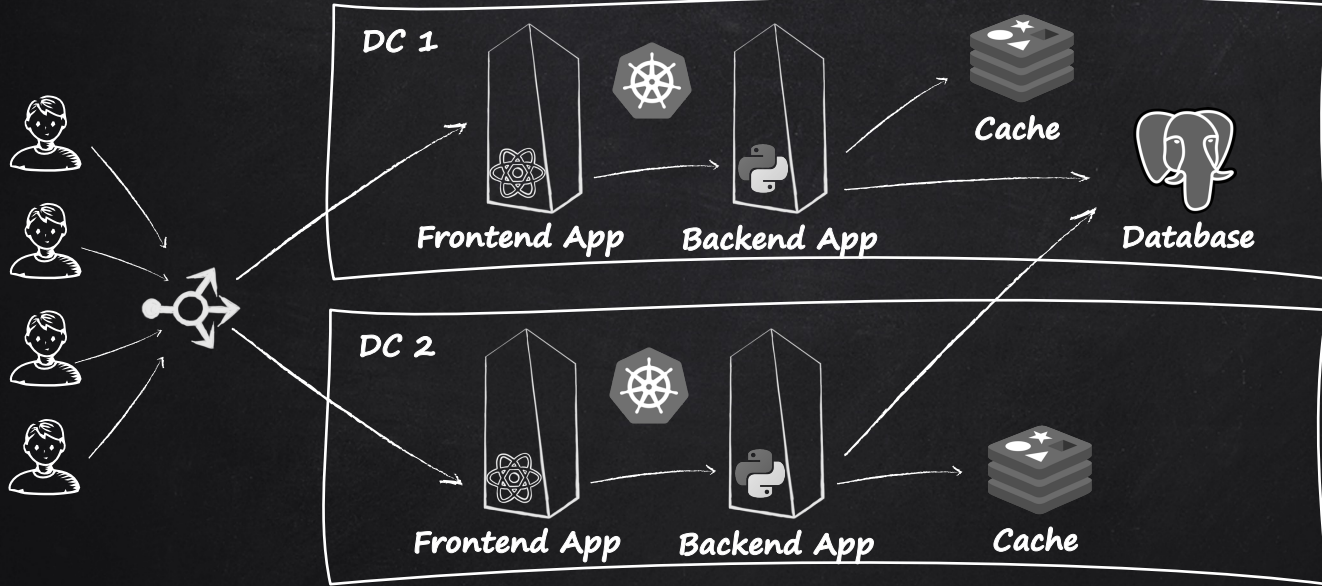


Описание регламента
собеседования



1

Обсуждение архитектуры



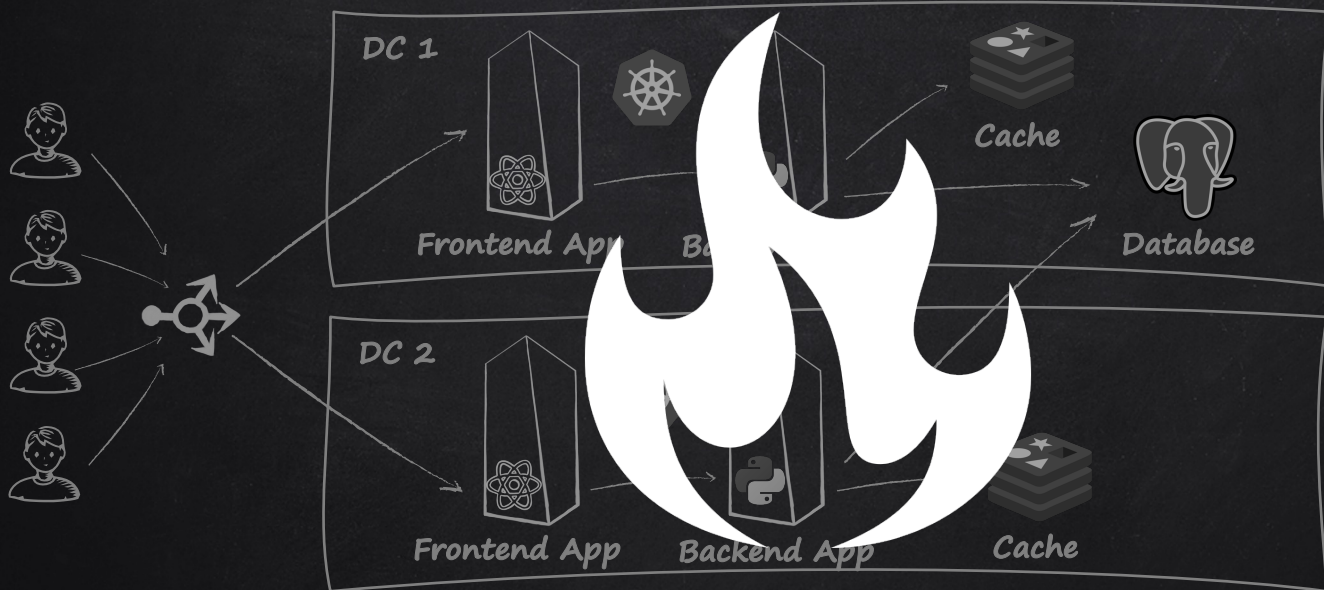


Описание регламента
собеседования



1

Обсуждение архитектуры



2

Старт инцидента
- Lead уехал в отпуск,
Jim остался дежурить
- Система внезапно
стало плохо

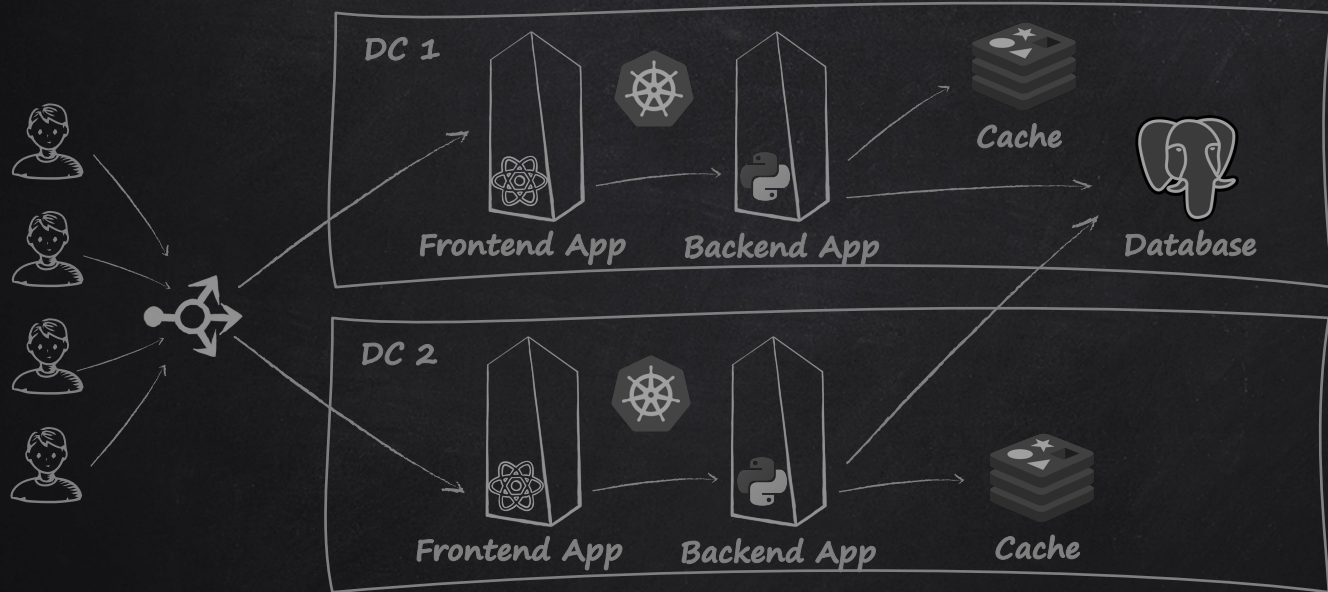
0

Описание регламента
собеседования



1

Обсуждение архитектуры



2

Старт инцидента
- Lead уехал в отпуск,
Jim остался дежурить
- Система внезапно
стало плохо

3

Диагностика
проблемы

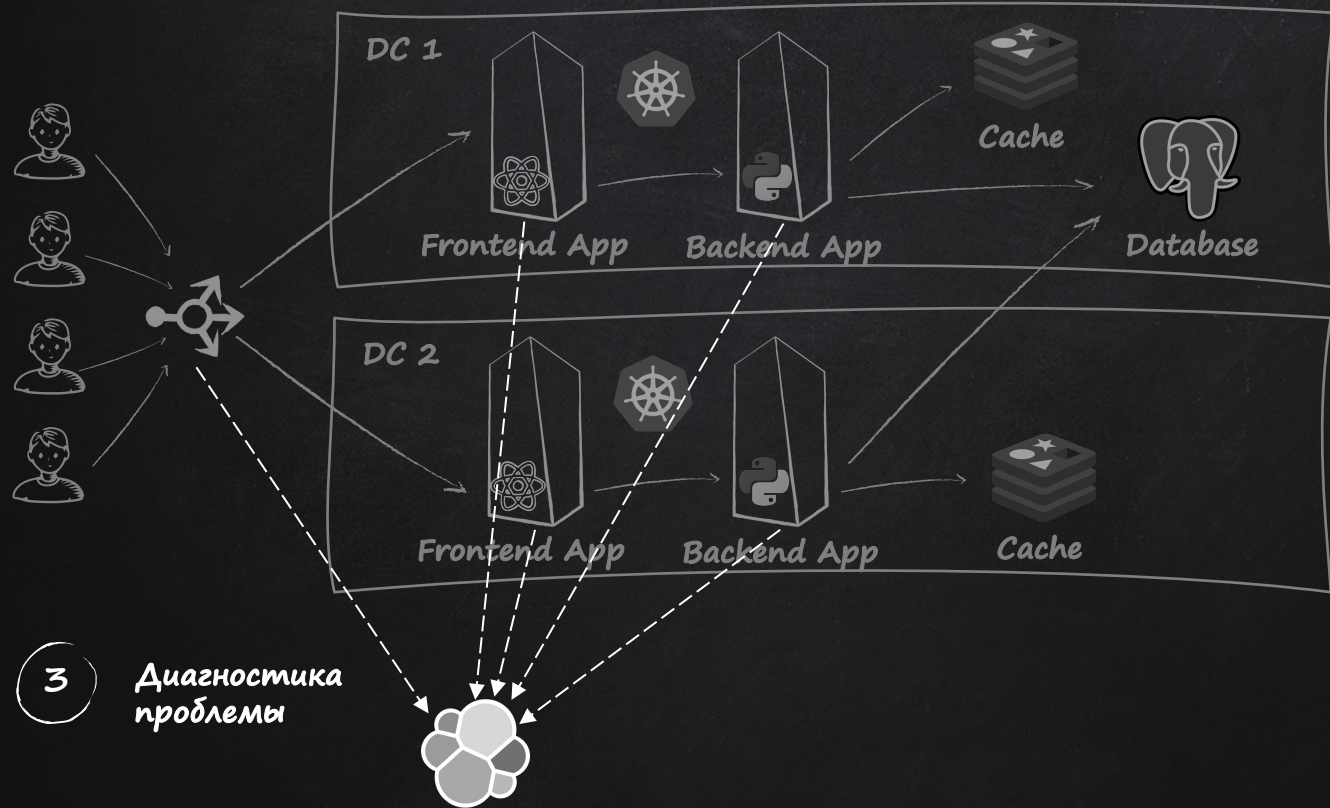
0

Описание регламента
собеседования



1

Обсуждение архитектуры



2

Старт инцидента
- Lead уехал в отпуск,
Jim остался дежурить
- Система внезапно
стало плохо

3

Диагностика
проблемы

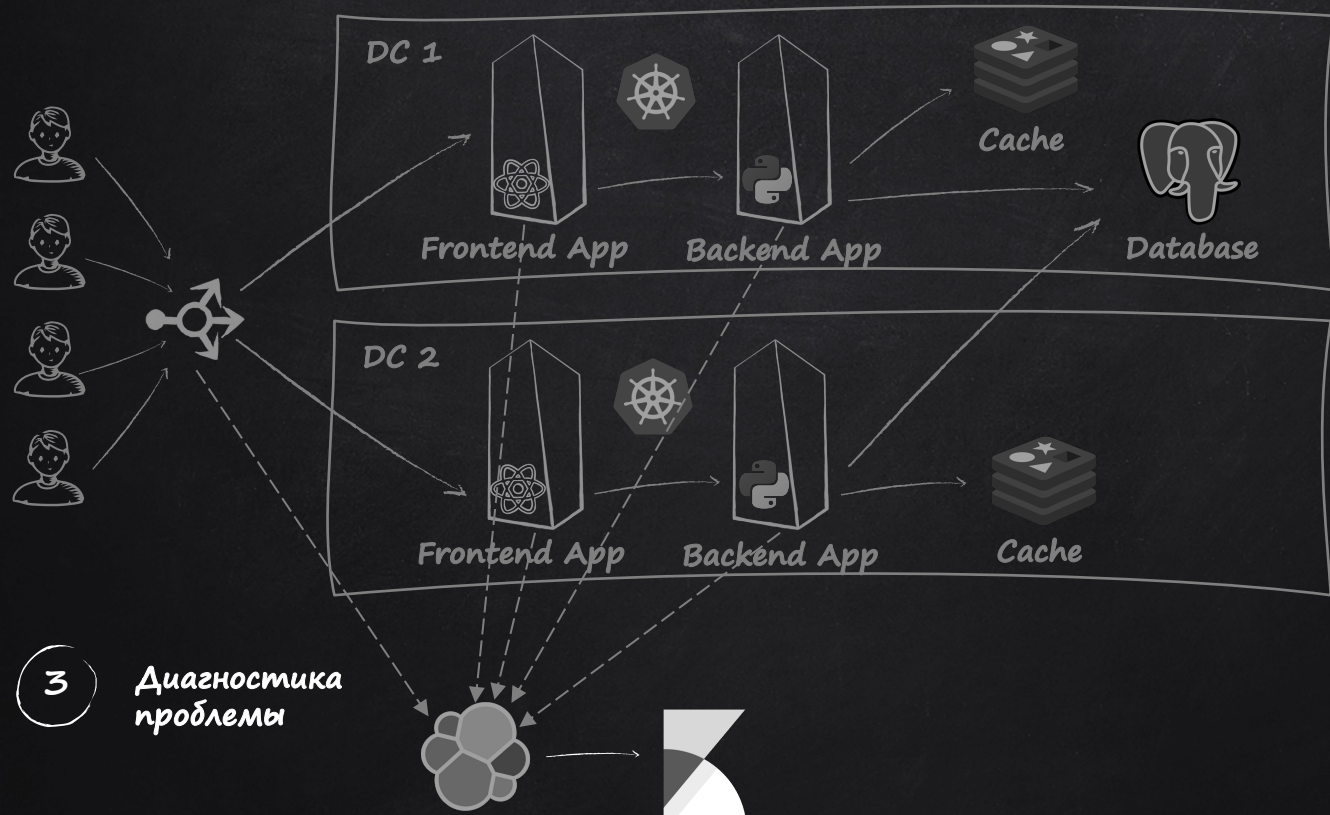
0

Описание регламента
собеседования



1

Обсуждение архитектуры



3

Диагностика
проблемы

2

Старт инцидента
- Lead уехал в отпуск,
Jim остался дежурить
- Система внезапно
стало плохо

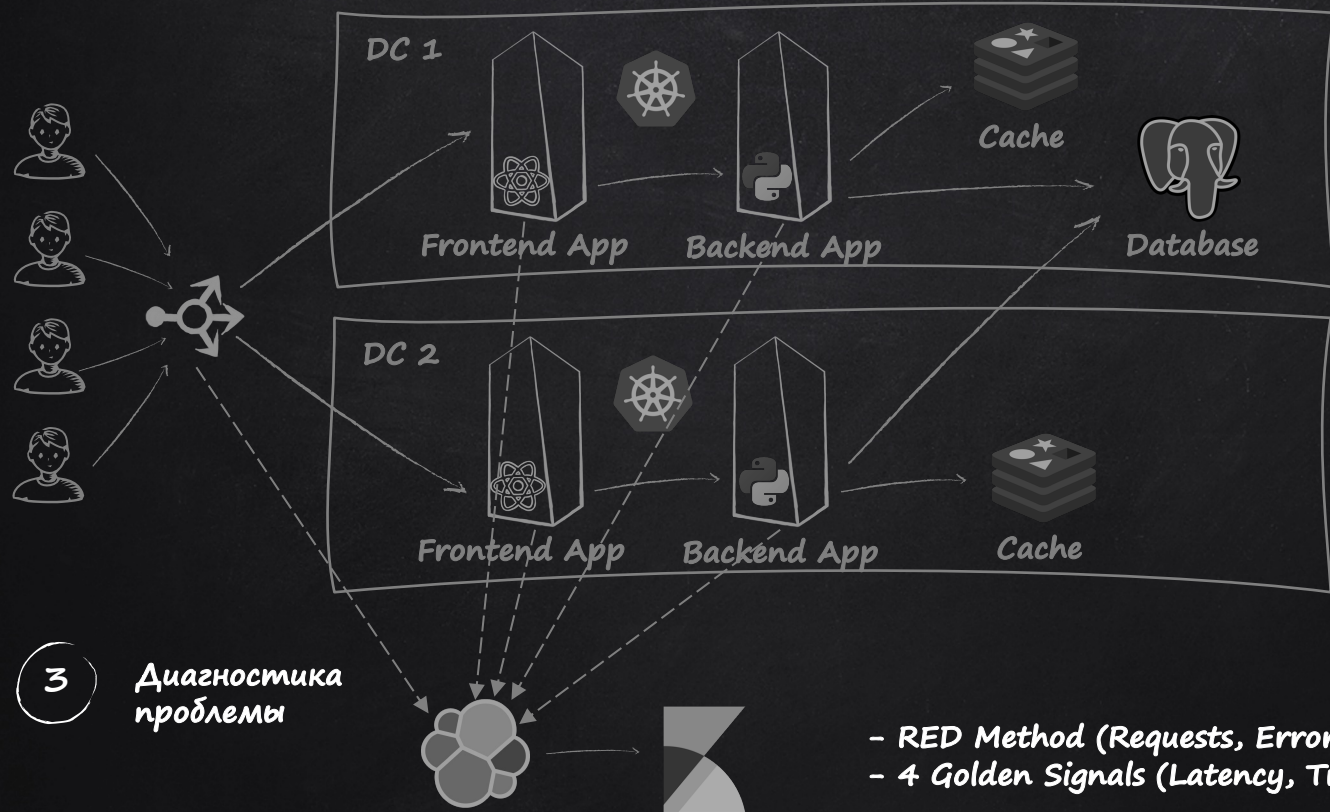
0

Описание регламента
собеседования



1

Обсуждение архитектуры



2

Старт инцидента
- Lead уехал в отпуск,
Jim остался дежурить
- Система внезапно
стало плохо

3

Диагностика
проблемы

- RED Method (Requests, Errors, Duration)
- 4 Golden Signals (Latency, Traffic, Errors, Saturation)

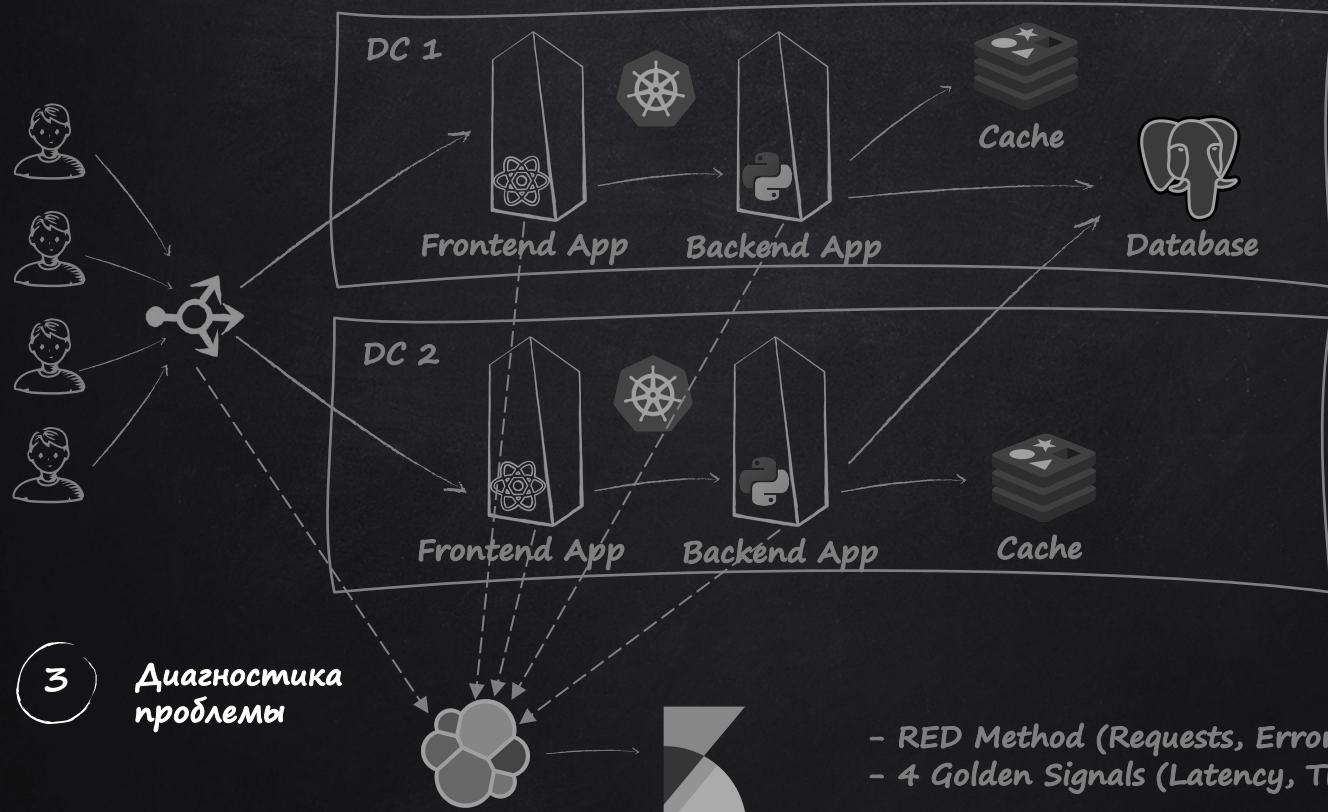
0

Описание регламента
собеседования



1

Обсуждение архитектуры



2

Старт инцидента
- Lead уехал в отпуск,
Jim остался дежурить
- Система внезапно
стало плохо

3

Диагностика
проблемы

Вырос процент 504 ошибок

- RED Method (Requests, Errors, Duration)
- 4 Golden Signals (Latency, Traffic, Errors, Saturation)

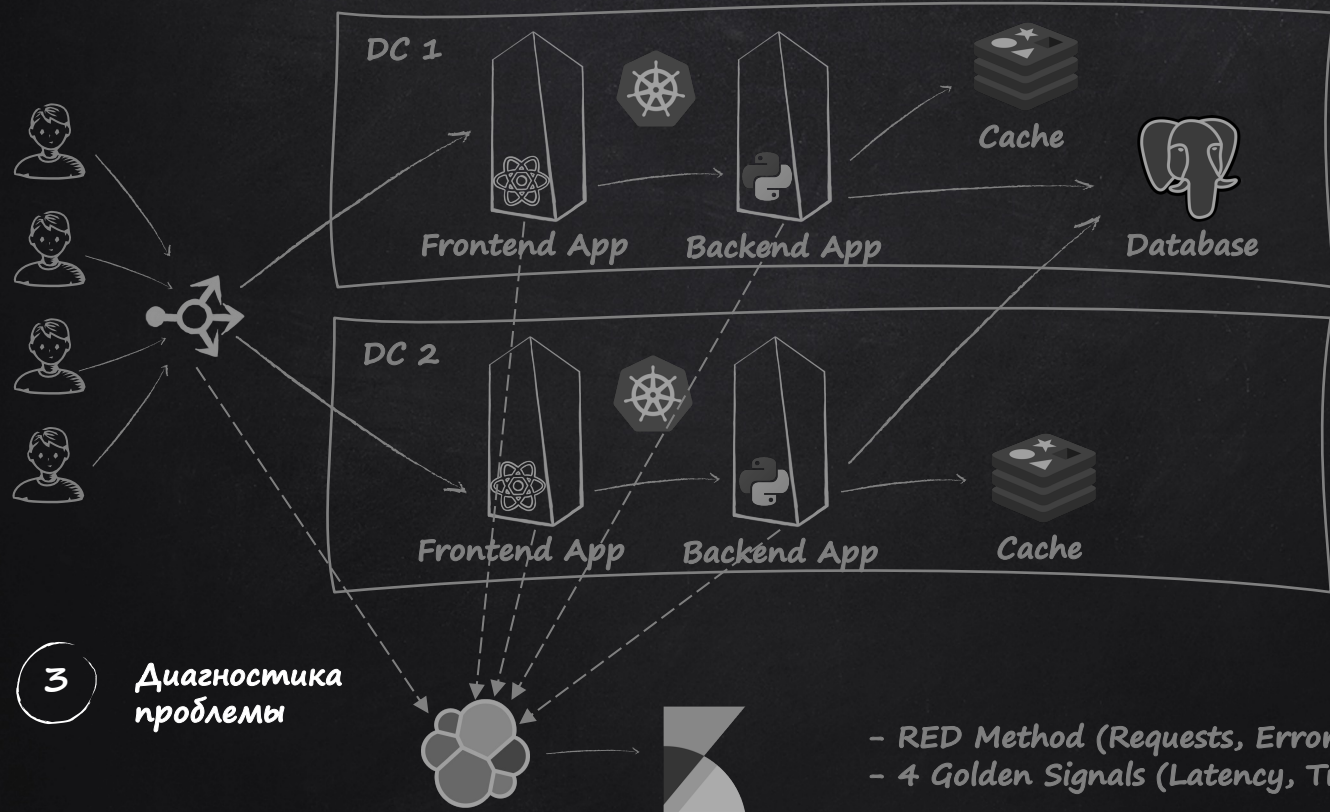
0

Описание регламента
собеседования



1

Обсуждение архитектуры



2

Старт инцидента
- Lead уехал в отпуск,
Jim остался дежурить
- Система внезапно
стало плохо

Gateway Timeouts, а значит
проверим логи приложения

Вырос процент 504 ошибок

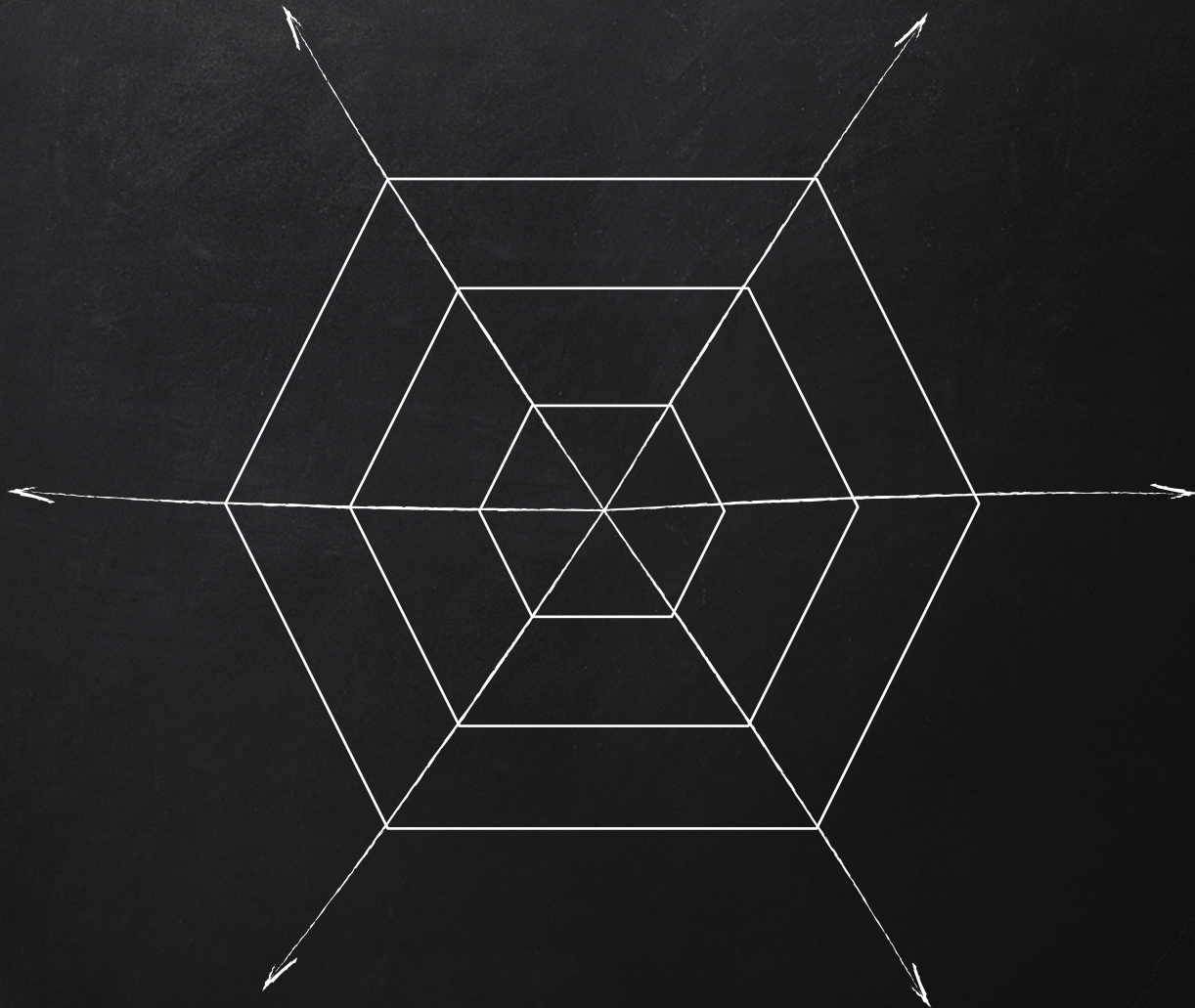
3

Диагностика
проблемы



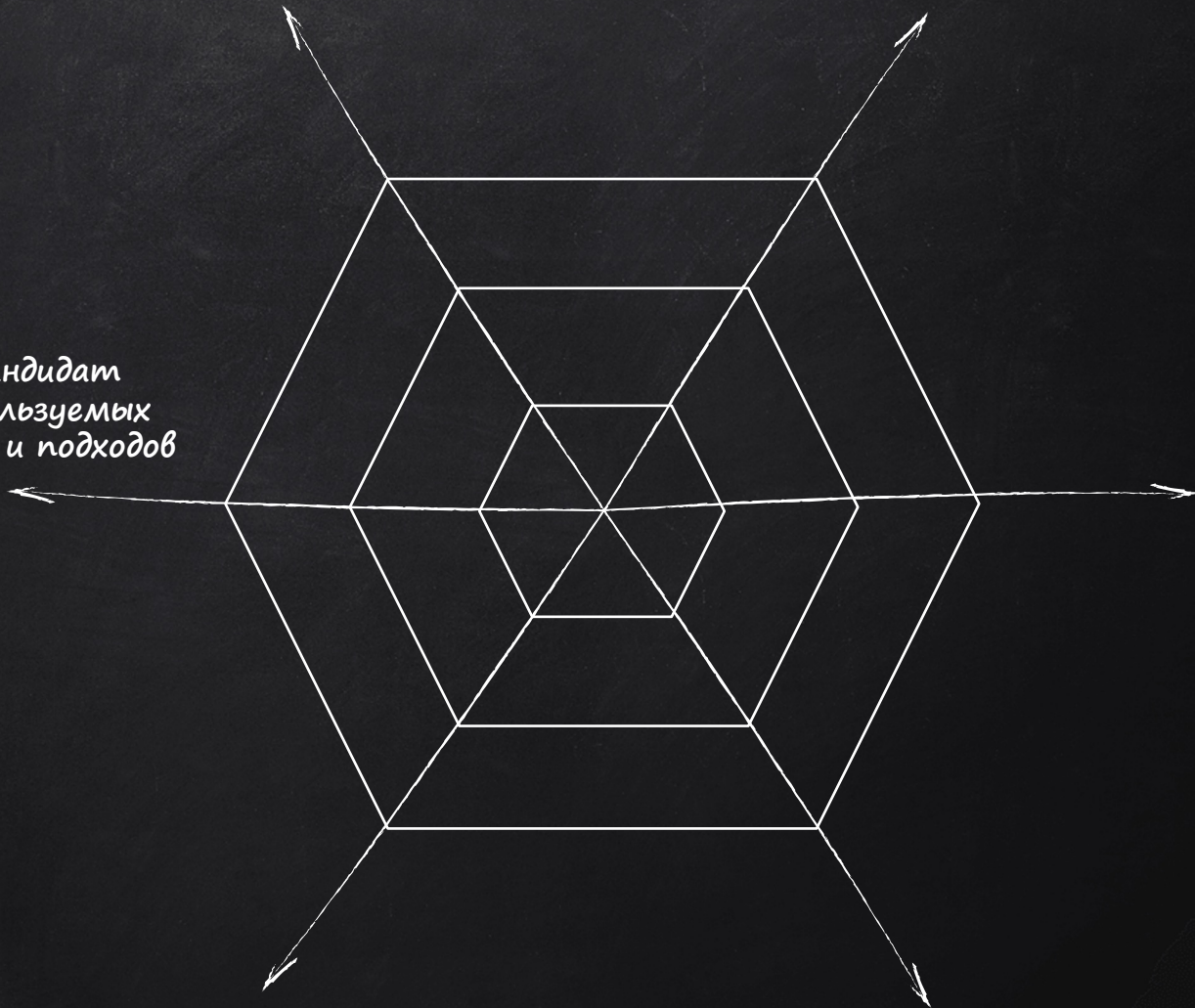
Как мы оцениваем
кандидата

Критерии оценки кандидатов



Критерии оценки кандидатов

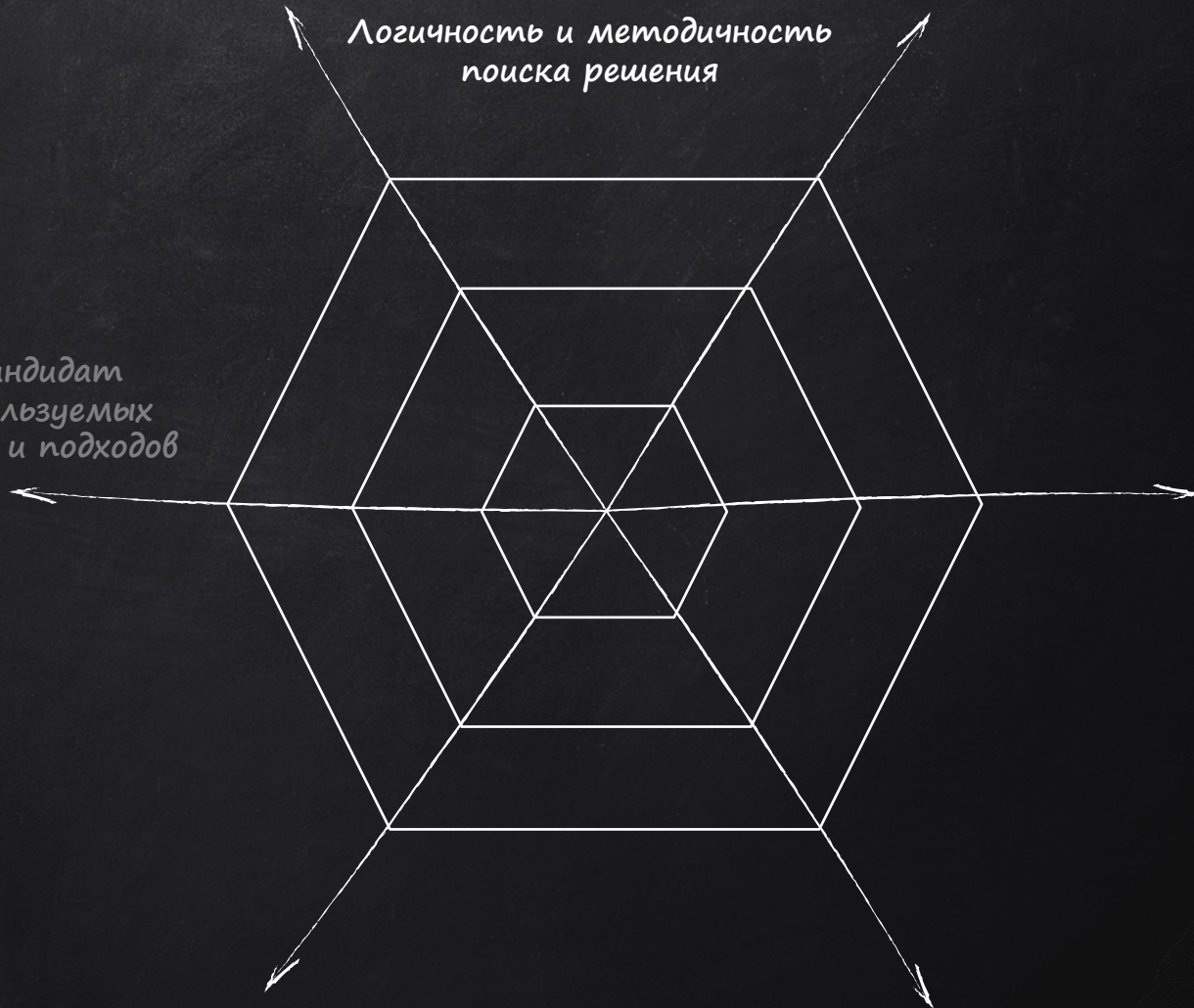
Кругозор кандидат
Широта используемых
инструментов и подходов



Критерии оценки кандидатов

Кругозор кандидат
Широта используемых
инструментов и подходов

Логичность и методичность
поиска решения

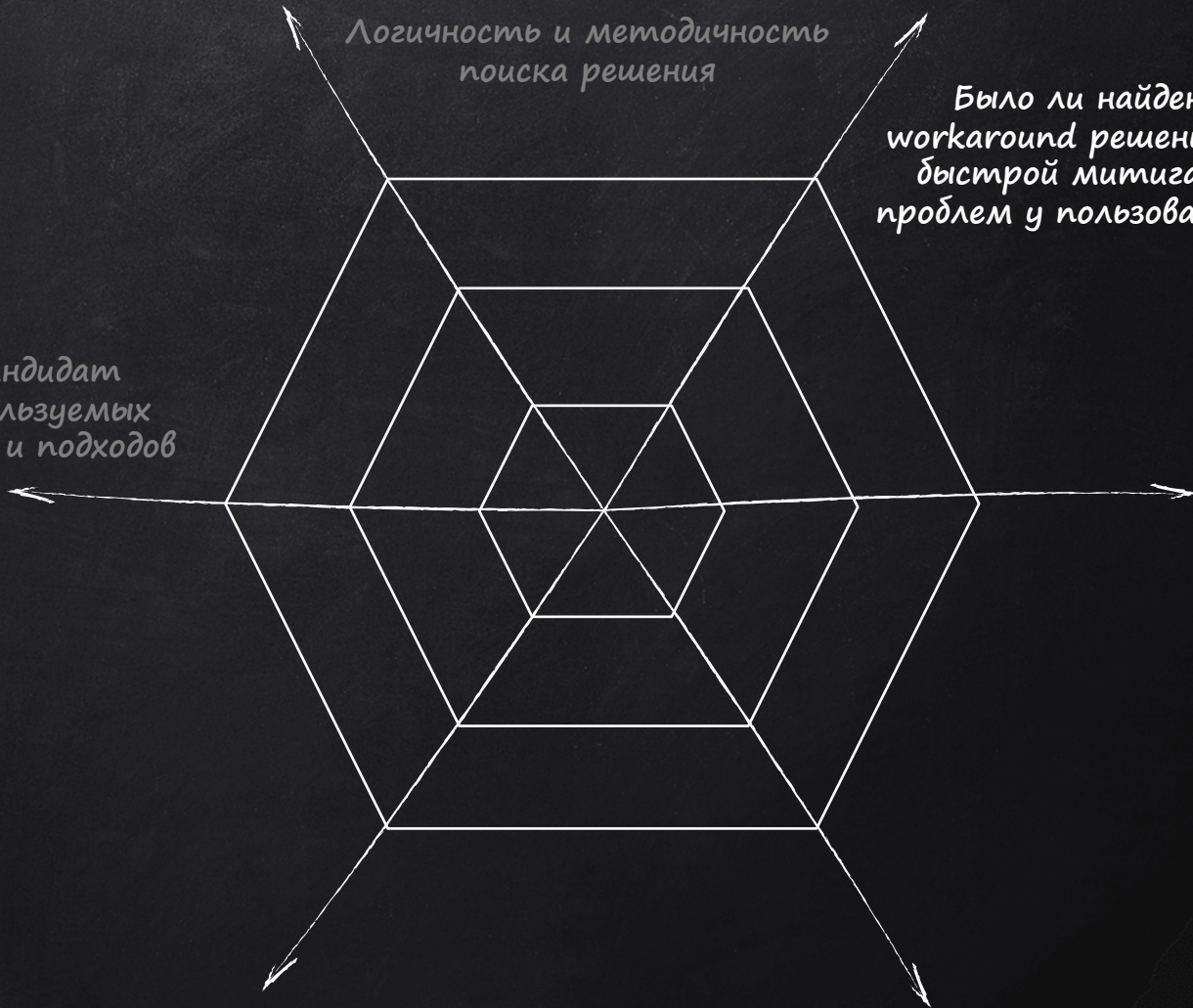


Критерии оценки кандидатов

Кругозор кандидат
Широта используемых
инструментов и подходов

Логичность и методичность
поиска решения

Было ли найдено
workaround решение для
быстрой митигации
проблем у пользователей



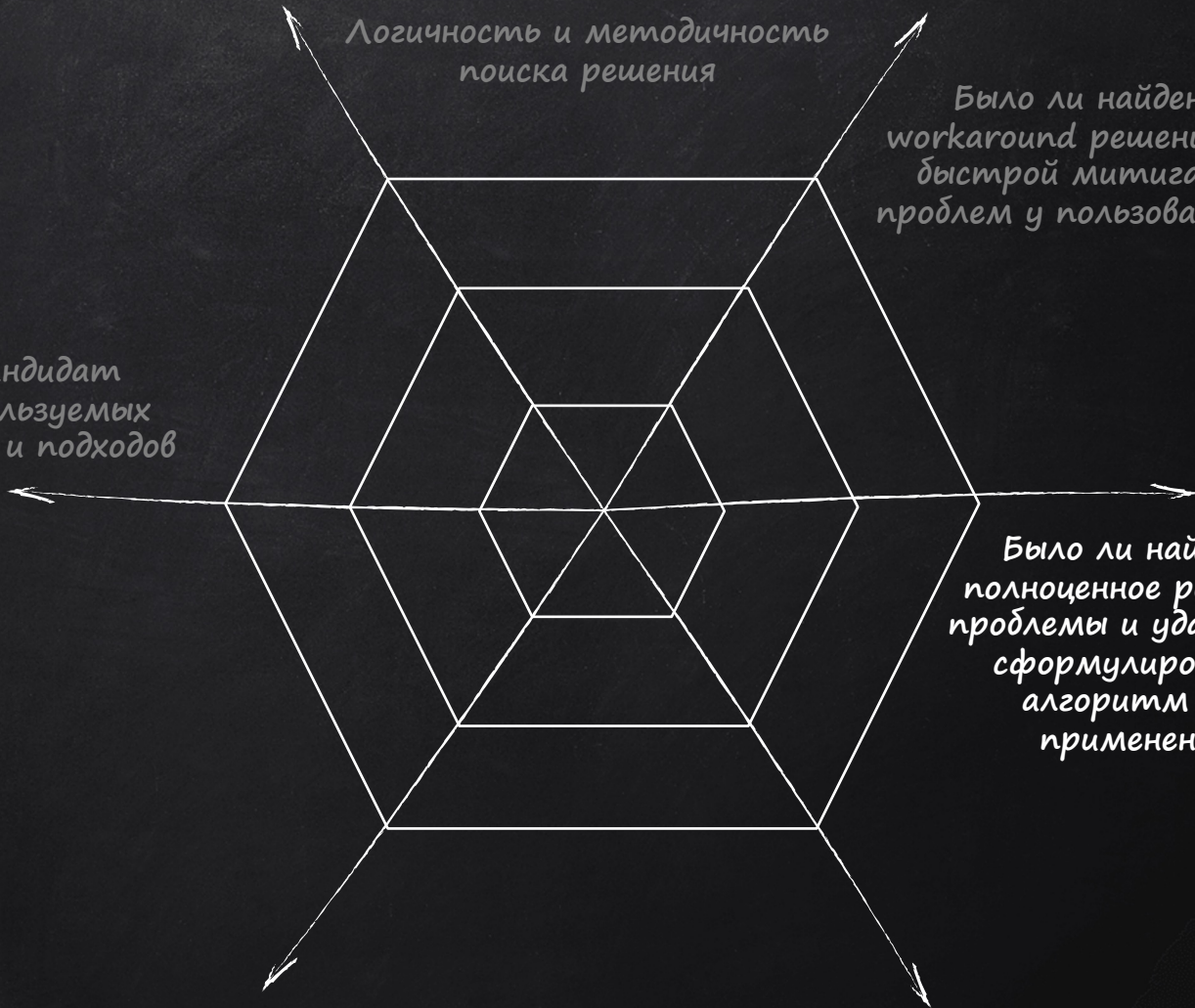
Критерии оценки кандидатов

Кругозор кандидат
Широта используемых
инструментов и подходов

Логичность и методичность
поиска решения

Было ли найдено
workaround решение для
быстрой митигации
проблем у пользователей

Было ли найдено
полноценное решение
проблемы и удалось ли
сформулировать
алгоритм его
применения



Критерии оценки кандидатов

Кругозор кандидат
Широта используемых
инструментов и подходов

Логичность и методичность
поиска решения

Было ли найдено
workaround решение для
быстрой митигации
проблем у пользователей

Было ли найдено
полноценное решение
проблемы и удалось ли
сформулировать
алгоритм его
применения

Раскопали ли root cause
проблемы, который объясняет
исходные симптомы и почему
алгоритм решения проблемы
сработал

Критерии оценки кандидатов

Кругозор кандидат
Широта используемых
инструментов и подходов

Предложены ли способы как
улучшить систему, чтобы
– Похожие проблемы не
повторялись в будущем
– Или как минимум мы о них
оперативно узнавали

Логичность и методичность
поиска решения

Было ли найдено
workaround решение для
быстрой митигации
проблем у пользователей

Было ли найдено
полноценное решение
проблемы и удалось ли
сформулировать
алгоритм его
применения

Раскопали ли root cause
проблемы, который объясняет
исходные симптомы и почему
алгоритм решения проблемы
сработал

Уровень кандидатов по этим осям

Кругозор кандидат
Широта используемых
инструментов и подходов

Предложены ли способы как
улучшить систему, чтобы
– Похожие проблемы не
повторялись в будущем
– Или как минимум мы о них
оперативно узнавали

Логичность и методичность
поиска решения

Было ли найдено
workaround решение для
быстрой митигации
проблем у пользователей

Было ли найдено
полноценное решение
проблемы и удалось ли
сформулировать
алгоритм его
применения

Раскопали ли root cause
проблемы, который объясняет
исходные симптомы и почему
алгоритм решения проблемы
сработал

Уровень кандидатов по этим осям

Кругозор кандидат
Широта используемых
инструментов и подходов

Предложены ли способы как
улучшить систему, чтобы
– Похожие проблемы не
повторялись в будущем
– Или как минимум мы о них
оперативно узнавали

Логичность и методичность
поиска решения

Было ли найдено
workaround решение для
быстрой митигации
проблем у пользователей

Было ли найдено
полноценное решение
проблемы и удалось ли
сформулировать
алгоритм его
применения

Раскопали ли root cause
проблемы, который объясняет
исходные симптомы и почему
алгоритм решения проблемы
сработал



Junior

Уровень кандидатов по этим осям

Кругозор кандидат
Широта используемых
инструментов и подходов

Предложены ли способы как
улучшить систему, чтобы
– Похожие проблемы не
повторялись в будущем
– Или как минимум мы о них
оперативно узнавали

Логичность и методичность
поиска решения

Было ли найдено
workaround решение для
быстрой митигации
проблем у пользователей

Было ли найдено
полноценное решение
проблемы и удалось ли
сформулировать
алгоритм его
применения

Раскопали ли root cause
проблемы, который объясняет
исходные симптомы и почему
алгоритм решения проблемы
сработал



Middle

Уровень кандидатов по этим осям

Кругозор кандидат
Широта используемых
инструментов и подходов

Предложены ли способы как
улучшить систему, чтобы
– Похожие проблемы не
повторялись в будущем
– Или как минимум мы о них
оперативно узнавали

Логичность и методичность
поиска решения

Было ли найдено
workaround решение для
быстрой митигации
проблем у пользователей



Senior

Было ли найдено
полноценное решение
проблемы и удалось ли
сформулировать
алгоритм его
применения

Раскопали ли root cause
проблемы, который объясняет
исходные симптомы и почему
алгоритм решения проблемы
сработал

Уровень кандидатов по этим осям

Кругозор кандидат
Широта используемых
инструментов и подходов

Предложены ли способы как
улучшить систему, чтобы
- Похожие проблемы не
повторялись в будущем
- Или как минимум мы о них
оперативно узнавали

Логичность и методичность
поиска решения

Было ли найдено
workaround решение для
быстрой митигации
проблем у пользователей



Senior+

Было ли найдено
полноценное решение
проблемы и удалось ли
сформулировать
алгоритм его
применения

Раскопали ли root cause
проблемы, который объясняет
исходные симптомы и почему
алгоритм решения проблемы
сработал

Пример профиля кандидата

Кругозор кандидат
Широта используемых
инструментов и подходов

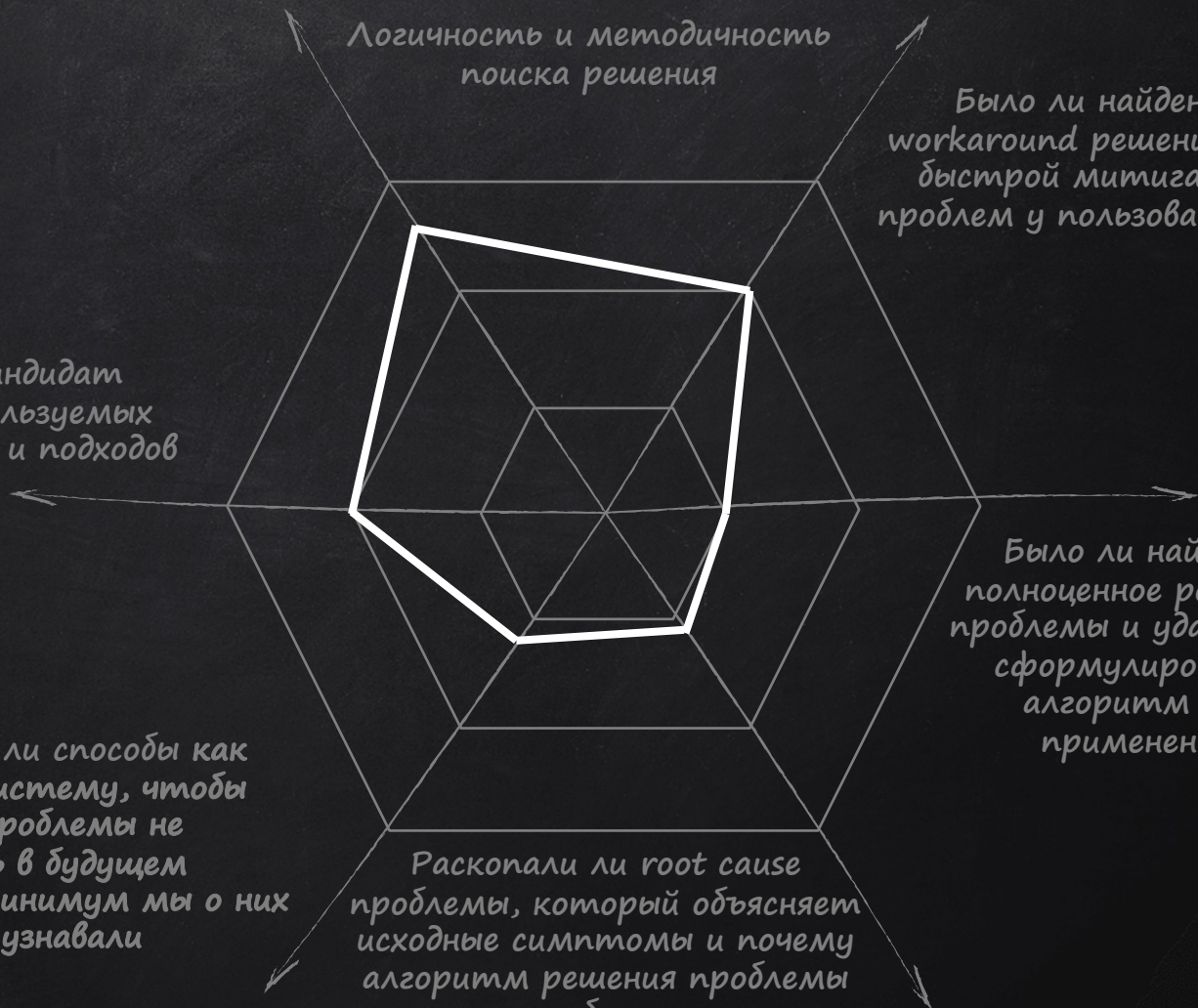
Предложены ли способы как
улучшить систему, чтобы
– Похожие проблемы не
повторялись в будущем
– Или как минимум мы о них
оперативно узнавали

Раскопали ли root cause
проблемы, который объясняет
исходные симптомы и почему
алгоритм решения проблемы
сработал

Логичность и методичность
поиска решения

Было ли найдено
workaround решение для
быстрой митигации
проблем у пользователей

Было ли найдено
полноценное решение
проблемы и удалось ли
сформулировать
алгоритм его
применения





Раньше для SRE мы проводили
и System Design, но потом
остался только
Troubleshooting. Почему?



Как выглядит дизайн секция

1

Название

Контекст

Требования:

- функциональные
- нефункциональные



1

Название

Контекст

Требования:

- функциональные
- нефункциональные



2

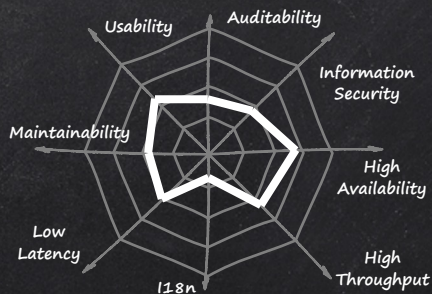
Формализация задачи:

- функциональность
- главные арх.

характеристики

Вопросы про сайзинг:

- users, requests, ...



1

Название

Контекст

Требования:

- функциональные
- нефункциональные



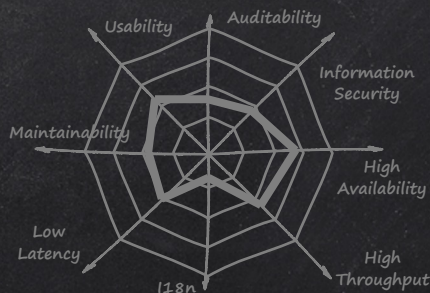
2

Формализация задачи:

- функциональность
- главные арх. характеристики

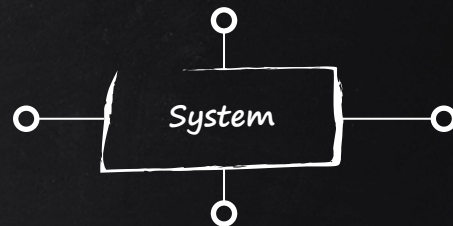
Вопросы про сайзинг:

- users, requests, ...



3

Границы системы
Обрабатываемые сценарии
Публичный API



1

Название

Контекст

Требования:

- функциональные
- нефункциональные



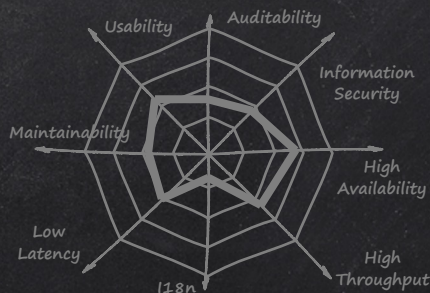
2

Формализация задачи:

- функциональность
- главные арх. характеристики

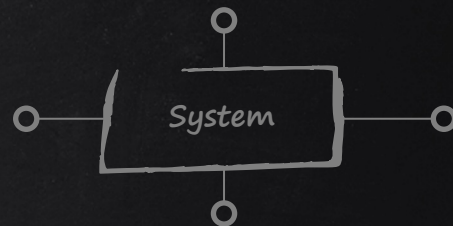
Вопросы про сайзинг:

- users, requests, ...



3

Границы системы
Обрабатываемые сценарии
Публичный API



4

Потоки данных и компоненты системы
Итеративная проработка:

- happy path
- нагрузка и нефункциональные требования
- corner cases, когда части системы отваливаются



1

Название

Контекст

Требования:

- функциональные
- нефункциональные



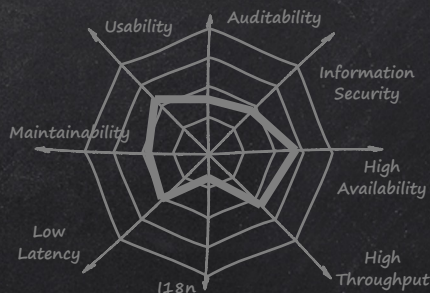
2

Формализация задачи:

- функциональность
- главные арх. характеристики

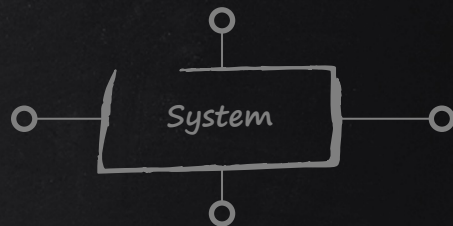
Вопросы про сайзинг:

- users, requests, ...



3

Границы системы
Обрабатываемые сценарии
Публичный API



4

Потоки данных и компоненты системы
Итеративная проработка:

- happy path
- нагрузка и нефункциональные требования
- corner cases, когда части системы отваливаются

5

Концептуальная схема готова:

- прописаны публичные API
- компоненты системы определены до классов систем (пример: k/v, rdbms, column family, etc)
- расписаны модели данных для обмена
- ясно что in-memory, а что persistent



1

Название

Контекст

Требования:

- функциональные
- нефункциональные



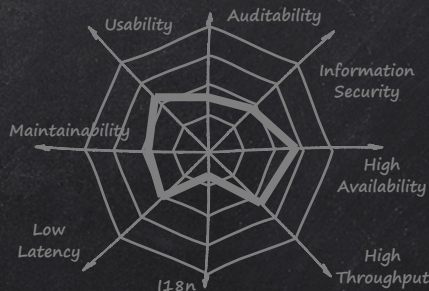
2

Формализация задачи:

- функциональность
- главные арх. характеристики

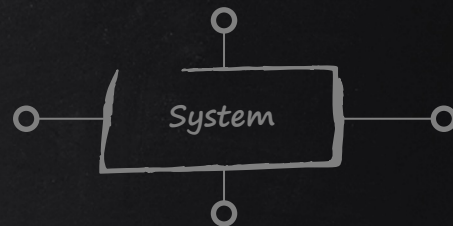
Вопросы про сайзинг:

- users, requests, ...



3

Границы системы
Обрабатываемые сценарии
Публичный API



4

Потоки данных и компоненты системы
Итеративная проработка:

- happy path
- нагрузка и нефункциональные требования
- corner cases, когда части системы отваливаются

6

Выбор конкретных технологий + расчет мощностей под нагрузку

5

Концептуальная схема готова:

- прописаны публичные API
- компоненты системы определены до классов систем (пример: k/v, rdbms, column family, etc)
- расписаны модели данных для обмена
- ясно что in-memory, а что persistent

Key-Value



Column-Family



Graph



Document



1

Название

Контекст

Требования:

- функциональные
- нефункциональные



6

Выбор конкретных технологий + расчет мощностей под нагрузку

Key-Value



Graph



Column-Family



Document



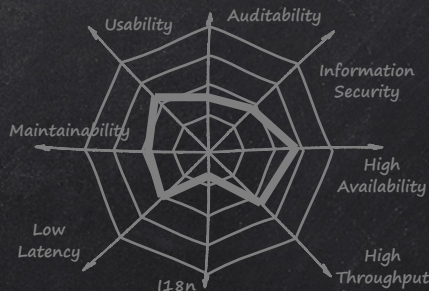
2

Формализация задачи:

- функциональность
- главные арх. характеристики

Вопросы про сайзинг:

- users, requests, ...



7

Дополнительные вопросы

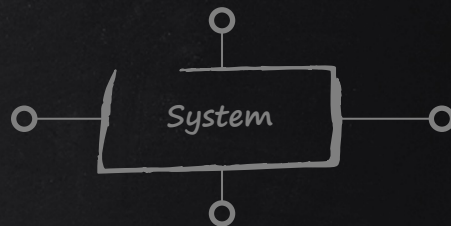
5

Концептуальная схема готова:

- прописаны публичные API
- компоненты системы определены до классов систем (пример: k/v, rdbms, column family, etc)
- расписаны модели данных для обмена
- ясно что in-memory, а что persistent

3

Границы системы
Обрабатываемые сценарии
Публичный API



4

Потоки данных и компоненты системы
Итеративная проработка:

- happy path
- нагрузка и нефункциональные требования
- corner cases, когда части системы отваливаются





VS



Сравним Troubleshooting и System Design



Troubleshooting

X На старте уже есть боевая система



System Design

X На старте только требования заказчика



Troubleshooting

- X На старте уже есть боевая система
- X Суть в том, чтобы потушить внезапно «загоревшуюся» систему



System Design

- X На старте только требования заказчика
- X Суть в том, чтобы спроектировать систему, которая не подвержена «возгоранием»



Troubleshooting

- X На старте уже есть боевая система
- X Суть в том, чтобы потушить внезапно «загоревшуюся» систему
- X Проверяем насколько системен подход к изучению симптомов, поиску временного лечения и дальше полноценному исцелению



System Design

- X На старте только требования заказчика
- X Суть в том, чтобы спроектировать систему, которая не подвержена «возгоранием»
- X Проверяем насколько системен подход к проектированию решения с учетом функциональных и нефункциональных требований и насколько жизнеспособная система получается в итоге



Troubleshooting

- X На старте уже есть боевая система
- X Суть в том, чтобы потушить внезапно «загоревшуюся» систему
- X Проверяем насколько системный подход к изучению симптомов, поиску временного лечения и дальше полноценному исцелению
- X Хорошо проходят ребята, которые близки к инфраструктуре и эксплуатации боевых систем



System Design

- X На старте только требования заказчика
- X Суть в том, чтобы спроектировать систему, которая не подвержена «возгоранием»
- X Проверяем насколько системный подход к проектированию решения с учетом функциональных и нефункциональных требований и насколько жизнеспособная система получается в итоге
- X Хорошо проходят ребята, которые близки к проектированию и разработке новой функциональности в сложных системах

В идеальном мире

SRE =

=



+



В идеальном мире



X Чтобы не просто тушить пожары, но и системно работать над надежностью систем

В идеальном мире



- X Чтобы не просто тушить пожары, но и системно работать над надежностью систем
- X Но реальный мир жесток и SRE кандидаты в среднем проходили System Design на уровень Junior, поэтому мы решили перестать спрашивать System Design

В идеальном мире



- Х Чтобы не просто тушить пожары, но и системно работать над надежностью систем
- Х Но реальный мир жесток и SRE кандидаты в среднем проходили System Design на уровень Junior, поэтому мы решили перестать спрашивать System Design
- Х Но решили учить этому наших сотрудников, как разработчиков, так и SRE



А как прокачаться в
Troubleshooting и
System Design



х Теория

- Изучить практики и подходы, например, SRE Book и SRE Workbook от Google
- Изучить инструменты для применения этих методов



х Теория

- Изучить принципы проектирования распределенных систем
- Изучить основные классы систем, их сильные стороны и границы применимости



х Теория

- Изучить практики и подходы, например, SRE Book и SRE Workbook от Google
- Изучить инструменты для применения этих методов

х Практика

- Работать в SRE команде над реальными системами
- Изучить публичные postmortems крупных и сложных систем
- Потренироваться в troubleshooting на задачах, сделанных по мотивам postmortems



х Теория

- Изучить принципы проектирования распределенных систем
- Изучить основные классы систем, их сильные стороны и границы применимости

х Практика

- Работать в роли архитектора над проектированием реальных систем
- Изучить архитектуру крупных сложных систем (от Google, Meta,...)
- Потренироваться в решении архитектурных ката

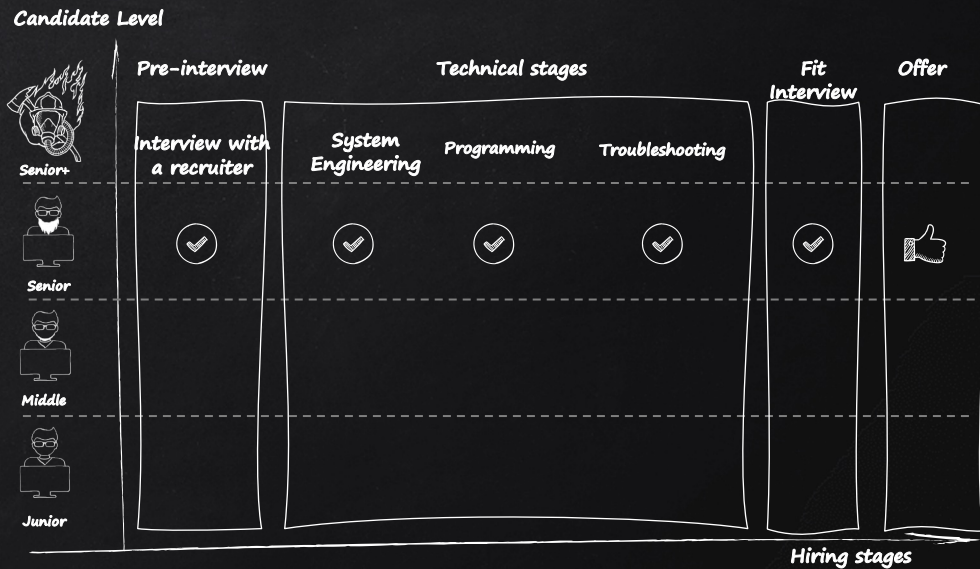


Подводим итоги



Мы обсудили вопросы

X Как выглядит процесс найма





Мы обсудили вопросы

- X Как выглядит процесс найма
- X Зачем мы проверяем навыки *troubleshooting* у кандидатов



Мы обсудили вопросы

- X Как выглядит процесс найма
- X Зачем мы проверяем навыки *troubleshooting* у кандидатов
- X Чем это полезно самому кандидату



Мы обсудили вопросы

- X Как выглядит процесс найма
- X Зачем мы проверяем навыки troubleshooting у кандидатов
- X Чем это полезно самому кандидату
- X Как выглядит секция troubleshooting



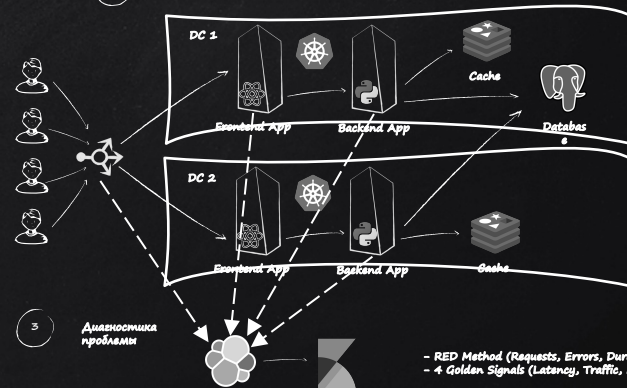


Мы обсудили вопросы

- Х Как выглядит процесс найма
- Х Зачем мы проверяем навыки troubleshooting у кандидатов
- Х Чем это полезно самому кандидату
- Х Как выглядит секция troubleshooting
- Х Как выглядит типовая задача

0 Описание регламента собеседования

1 Обсуждение архитектуры



2 Старт инцидента
- Lead ушел в отпуск,
Jim остался дежурным
- Система внезапно
стала плохо

Gateway Timeouts, а значит
проверим логи приложения

Вырос процент SLO ошибок

3 Диагностика
проблемы

- RED Method (Requests, Errors, Duration)
- 4 Golden Signals (Latency, Traffic, Errors, Saturation)



Мы обсудили вопросы

- Х Как выглядит процесс найма
- Х Зачем мы проверяем навыки *troubleshooting* у кандидатов
- Х Чем это полезно самому кандидату
- Х Как выглядит секция *troubleshooting*
- Х Как выглядит типовая задача
- Х Как мы оцениваем кандидатов





Мы обсудили вопросы

- X Как выглядит процесс найма
- X Зачем мы проверяем навыки *troubleshooting* у кандидатов
- X Чем это полезно самому кандидату
- X Как выглядит секция *troubleshooting*
- X Как выглядит типовая задача
- X Как мы оцениваем кандидатов
- X Как выглядит секция *system design*

1

Навыки
Кандидата
Требования
– функциональные
– нефункциональные



6

Выбор конкретных
технологий + расчет
мощностей под
нагрузку



2

Формализация задачи:
– функциональность
– главные арг.
характеристики
Вопросы про сценарии:
– users, requests, ...



7

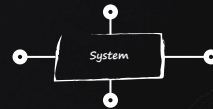
Дополнительные
вопросы

5

Концептуальная схема готова:
– прописали публичный API
– компоненты системы определяем до
классов системы (пример: k/v, rdbs, column
family, etc)
– расписали модели данных для обмена
– ясно что in-memory, а что persistent

3

Границы системы
Обрабатываемые сценарии
Публичный API



4

Потоки данных и компоненты системы
Итеративная проработка:
– happy path
– нагрузка и нефункциональные требования
– corner cases, когда части системы
отваливаются





Мы обсудили вопросы

- X Как выглядит процесс найма
- X Зачем мы проверяем навыки *troubleshooting* у кандидатов
- X Чем это полезно самому кандидату
- X Как выглядит секция *troubleshooting*
- X Как выглядит типовая задача
- X Как мы оцениваем кандидатов
- X Как выглядит секция *system design*
- X В чем отличия *troubleshooting* и *system design*



Troubleshooting

На старте уже есть боевая система

Суть в том, чтобы потушить внезапно «загоревшуюся» систему

Проверяем насколько системен подход к изучению симптомов, поиску временного лечения и дальше полноценному излечению

Хорошо проходят ребята, которые близки к инфраструктуре и эксплуатации боевых систем



System Design

На старте только требования заказчика

Суть в том, чтобы спроектировать систему, которая не подвержена «возгораниям»

Проверяем насколько системен подход к проектированию решения с учетом функциональных и нефункциональных требований и насколько жизнеспособная система получается в итоге

Хорошо проходят ребята, которые близки к проектированию и разработке новой функциональности в сложных системах



Мы обсудили вопросы

- X Как выглядит процесс найма
- X Зачем мы проверяем навыки troubleshooting у кандидатов
- X Чем это полезно самому кандидату
- X Как выглядит секция troubleshooting
- X Как выглядит типовая задача
- X Как мы оцениваем кандидатов
- X Как выглядит секция system design
- X В чем отличия troubleshooting и system design
- X Как прокачаться в system design и troubleshooting



Теория

Изучить практики и подходы, например, SRE Book и SRE Workbook от Google
Изучить инструменты для применения этих методов

Практика

Работать в SRE команде над реальными системами
Изучить публичные postmortems крупных и сложных систем
Потренироваться в troubleshooting на задачах, сделанных по мотивам postmortems



Теория

Изучить принципы проектирования распределенных систем
Изучить основные классы систем, их сильные стороны и границы применимости

Практика

Работать в роли архитектора над проектированием реальных систем
Изучить архитектуру крупных сложных систем (от Google, Meta,...)
Потренироваться в решении архитектурных задач



Ссылки для
дальнейшего изучения



Пожалуйста,
оцените доклад



Источники знаний

- Статья по мотивам этого доклада
 - bit.ly/SREInterview
- Материалы про Troubleshooting
 - Google SRE Book – bit.ly/GoogleSREBook
 - SRE Workbook – bit.ly/GoogleSREWorkbook
 - Обзор Mobile SRE Report – bit.ly/mobileSREbook
 - Book "Release It" (second edition)
 - Github Repo with list of postmortems – github.com/danluu/post-mortems
- Материалы про System Design
 - Мои статьи
 - про System Design Interview – bit.ly/sysDesign
 - про подготовку к System Design – bit.ly/prepSysDesign