

27–28 мая 2023

Новосибирск, Экспоцентр
1 000 участников

The logo for CodeFest, featuring the words "Code" and "Fest" stacked vertically in white, bold, sans-serif font, enclosed within a black speech bubble shape. The speech bubble is set against a larger, light blue speech bubble background that points towards the top right corner of the page.

**Code
Fest**

**Темы, которые
ХОТИМ услышать
на CodeFest 13**

Management

Ниже список тем и направлений, о которых было бы интересно услышать. Но это не значит, что другие темы нам совсем не интересны — management — очень широкое направление.

Про вызовы времени

Резидентное лидерство — как в хрупком и нестабильном мире управлять и работать с командой. Систематизация опыта релокации — команд, компаний. Как выстраивать работу в новой реальности.

Про команду

Доверие в команде и компании — как его выстраивать на разных уровнях в работе. Как оценить эффективность команды? На какие метрики смотреть и с какой периодичностью? Какие инструменты использовать для сбора метрик команд? Влияние удалёнки на команду и работу внутри неё. Модели и метрики команды. Onboarding на удалёнке. Как создавать и поддерживать крутые команды? Типы лидерства, типы команд: какие сочетания работают лучше? Рoadmap команды: как ведёте, какие механики задействуете, эффективно ли? Новые подходы и инструменты в командной работе.

Личное развитие и карьера

Как расти и выстроить свой профессиональный путь? Как не выгореть в профессии? Что делать, если текущая работа уже не «торт»? Как понять, что нужен ментор, и как его найти? Выход из IT: есть ли жизнь после IT? Менторство. Как сделать процесс менторства эффективным: взгляды со стороны как менторов, так и менти.

Про и для Лидов

Реальная цель Лида: что должен и чего не должен делать. А нужен ли вообще TeamLead? За что он ответственен и как оценить Лида? Как развивать сеньористых сеньоров, если ты Лид? Чего боятся начинающие Лиды? Где им черпать вдохновение и какие навыки нужно прокачивать?

Коммуникации

Как решаются сложности в коммуникациях внутри команды? Как выстроить коммуникации с заказчиками и топ-менеджерами. Конфликты внутри команды. Как слышать и слушать всех участников? Эмпатия менеджера: как лучше понимать чувства, ожидания и мотивы команды и обычных пользователей.

Современные и альтернативные подходы к управлению

Работаете в бирюзовой компании — расскажите, как вам живётся? Ваш опыт работы и внедрения принципов самоуправления в компании. Ваша команда работает по «настоящему» Scrum — поделитесь опытом, как вам удаётся соблюдать все ритуалы и к каким результатам это привело? Как вы масштабируете успешные кейсы на всю компанию?

Истории про стартапы

Как живут стартапы внутри больших корпораций. Как вырастить pet-проект, чтобы он превратился в настоящий бизнес? Как пройти отбор в международных акселераторах? Твоя история pet-проекта и во что она переросла по итогу.

Про процессы

Эволюция Upstream. Управление входящими запросами от идеи и до подготовки к реализации. Как эффективно управлять процессом разработки продукта. Как вы упорядочиваете процессы? Как вы решаете вопросы прогнозирования сроков? Различные эксперименты с подходами к разработке: почему вам «зашло» и от чего, наоборот, отказались? Роль менеджера проектов и её трансформация.

Про продукты

Стратегия. Вывод продукта на рынок. Развитие и рост продукта. Исследования. Монетизация. Метрики. Инструменты и фреймворки для создания продуктов.

Как создавать полезные и эффективные продукты для бизнеса? Продуктовая культура в компании и роль продакт-менеджера. Успешные продукты и истории «когда не полетело».

Экспериментальные темы

Ментальное здоровье. Как быстро и эффективно учиться? Осознанность как способ нахождения баланса в жизни и на работе.

Web 3

Новая секция, посвященная Web 3, технологиям блокчейна и всему, что с ними связано. Индустрия растет, взлетает, падает, и на этом фоне появляется всё больше проектов, которые реально соприкасаются с бизнесом, государствами и жизнью людей. Некоторые из тем, которые мы планируем затронуть на грядущем Codefest'e:

Знакомство с Web 3 и блокчейном для новичков

Обзор основных концепций Web 3, включая технологию распределенной книги записей, работу с smart-контрактами, механизмы консенсуса и экономику токенов.

Проблемы производительности блокчейн сетей

Обсудим проблемы, связанные с обработкой большого объема транзакций и данных в блокчейн сетях, а также поиск решений для повышения их производительности и масштабируемости.

Безопасность и защита данных

Вопросы безопасности и защиты данных в блокчейн сетях, включая темы такие как хеширование, криптография, конфиденциальность и аутентификация.

Использование и адаптация блокчейн технологии в реальных бизнес-сценариях

Примеры использования блокчейн технологии в различных отраслях и индустриях, а также процессы и инструменты, которые необходимы для успешной адаптации и использования блокчейн технологии в бизнесе.

Применение NFT в искусстве, коллекционировании и коммерческой сфере

Как NFT технология меняет индустрию искусства, коллекционирования и коммерческую сферу, включая различные применения, возможности и сложности

Backend

Языки программирования и фреймворки

Куда эволюционируют языки программирования? Насколько они удобны для создания нового кода микросервисов, serverless-программирования, или текущий подход только генерирует тонны бойлерплейт-кода и вызывает распухание популярных фреймворков?

Какой язык наиболее выразителен для своей сферы? Насколько выбор языка программирования влияет на стоимость владения и поддержки? Куда вообще сейчас «двигаются» языки программирования и фреймворки? Появляются ли принципиально новые инструменты, устроенные иначе, чем те, которые мы привыкли использовать? Поделитесь с нами своим опытом или исследованиями на данные темы.

.NET

Замеряли производительность? Копались в новом низкоуровневом API? Планируете оптимизацию под новые фишки? Наши участники любят хардкор. Что нового для enterprise-разработчиков? Присматриваетесь к HTTP/3? Языки и экосистема динамично развиваются. Можете рассказать про свой опыт адаптации к новому?

JVM

Что нового в мире Java/Kotlin/Scala? Spring по-прежнему актуален или будет вытеснен другими фреймворками? Какими и почему? Возможно, будущее энтерпрайз-разработки за Spring Native? Или Quarkus — наше всё? Успели пощупать GraalVM? Как впечатления от новой Java? Чего ждёте от будущих версий Java?

Go

Уже пробовали использовать дженерики или ждёте полноценного выхода? Возможно, знаете какие-то сценарии, в которых дженерики на порядок улучшают или ухудшают ситуацию? Используете ли вы Go как-то нестандартно — для мобильной и/или фронтенд-разработки? Как переход на Go улучшил или ухудшил вашу жизнь? Расскажите нам о своём необычном опыте в разработке на языке программирования Go.

Python

Недавно Python вышел на первое место в рейтинге TIOBE. Настолько ли он идеален? Куда развиваться дальше? Что нового в экосистеме? Уже используете Pattern Matching? Решены ли проблемы с производительностью, за которые конкуренты ругают язык? Жаждем новой информации об одном из самых популярных языков в мире.

Современные тренды в разработке

Сервисы, технологии и подходы:

- решающие проблемы реального мира: боремся с пандемией (медтех и проблемы удалёнки);
- с возрастающими требованиями пользователей (например, доставляем за 15 минут);
- меняющие привычки (к примеру, каршеринги, кикшеринги, блокчейн, супераппы);
- оцифровывающие офлайн-бизнес — там, куда ещё не ступал байт компьютера.

О чём сейчас говорят в сообществе разработчиков? Кому перейдут в наследство микросервисы? Как сейчас общаются приложения? Какие новые вызовы появляются?

No-Code/Zero-Code

Много говорят о no-code/zero-code. Пока не очень понятно, несёт ли это пользу для нас, разработчиков или, наоборот, лишает нас работы?

Serverless

Очередной buzzword или следующая парадигма? Возможно, у вас есть успешный проект с таким подходом? Или, наоборот, что-то пошло не так и пришлось откатывать всё назад? Приходите, уверены, что всем будет интересно послушать ваш опыт!

AI/ML для разработки

Скоро ли инструменты вроде Github CoPilot или TabNine смогут приносить реальную пользу в разработке? Можно ли разработчиков избавить от рутины и доверить некоторые задачи алгоритмам? Или в этом случае станет пора искать новую работу? С удовольствием послушаем, как поумнели машины и помогают нам разрабатывать лучшие продукты.

Проектирование и архитектура

Меняются ли подходы к проектированию архитектуры в последние годы? Возможно, вы узнали или выработали для себя новые паттерны или подходы, новые способы решения актуальных задач? Возможно, расскажете нам об унификации кода сервисов в вашей организации — как жить и версионировать внутренний фреймворк? Или вы можете поведать нам истории успехов или провалов в архитектуре ваших продуктов и рассказать о сделанных выводах?

Enterprise-разработка

Решаете сложные задачи бизнеса и вам удаётся сохранять высокий темп поставки, обеспечивать хорошее качество? Расскажите про свой подход к разработке, про архитектуру, которая оказалась удобной для решения вашего класса бизнес-задач. Мы бы хотели услышать про практические кейсы, что сработало, а что — нет. Какие технологии и инструменты помогли в ускорении разработки и повышении качества, скорости поставки.

Интеграция

Количество систем и технологий растёт по экспоненте: монолиты, микросервисы, serverless — часто в рамках одной компании соседствует огромное количество систем из разных эпох. Возможно, вы нашли способ сделать удобную интеграцию между ними? Скорее всего, на пути к идеальной картине были сложности и интересные истории, которыми стоит поделиться с коллегами по цеху.

Оптимизация производительности

Есть интересные кейсы, как вы что-либо наладили и это дало толчок к ускорению? Или у вас есть опыт, который помог значительно сэкономить на железе за счёт оптимизации по объёму данных в оперативной или постоянной памяти? Расскажите о своём опыте оптимизации производительности на примере ваших продуктов.

Алгоритмы и структуры данных

Что-то знаете об интересных алгоритмах или структурах данных, появившихся в различных инструментах или внедрённых у вас? Используете какие-то нестандартные подходы к распределённым вычислениям или синхронизации данных? Давайте похарадкорим на конференции!

Высокая нагрузка и масштабирование

Эффект пандемии показал, что нужно в любой момент быть готовым к кратному увеличению нагрузки на сервисы. Нам интересно всё, что поможет слушателям бороться с нагрузкой: истории, опыт, подходы, инструменты. Продвинутый уровень: честно рассказать про историю падения или деградации сервиса и о том, как вы это побороли и какие выводы сделали.

*Ops и инфраструктура

В нашей жизни всё больше слов с суффиксом *Ops — DevOps/GitOps/DevSecOps/FinOps/ЧтоТоOps — это просто уловки маркетологов или они действительно что-то качественно меняют в нашей работе?

Observability

Метрики, системы мониторинга, дашборды, трассировка, логи и системы журналирования, алерты и всё, что поможет открыть глаза на происходящее в продакшене.

Chaos Engineering

Построена большая распределённая инфраструктура, все приложения кластеризованны, базы реплицируются и бекаются, но так ли всё надёжно, как кажется? Расскажите, как вы тестируете отказоустойчивость в красках.

CI Templates

Современная доставка приложения — пайплайн на сборку образа, ещё один на поднятие инстанса, потом допишем на тесты и проверку безопасности, теперь деплоим в кубер, а у нас же ещё serverless есть — о боже, как много Boilerplate... Как не утонуть в тоннах пайплайнов и переиспользовать код. Можно ли использовать в CI/CD подходы, которые мы используем в обычной разработке кода?

GitOps

Используете GitOps? Насколько данная практика подходит вашей компании? Какие проблемы были в процессе внедрения и как их решили? А может быть, вам эта история не подошла? Насколько сочетание подходов GitOps и использование kubernetes-операторов улучшило или ухудшило ваши процессы производства? Расскажите, где без него не обойтись, а когда его лучше не использовать. Кейсы/паттерны как угробить свои репозитории.

DevSecOps

Всё об организации безопасной разработки — от инструментов до изменений в процессах производства.

Технологии хранения и передачи данных

Расскажите нам всё о технологиях хранения и передачи данных: СУБД, распределённые кэши, очереди, SQL, NewSQL, NoSQL, файловые хранилища, протоколы передачи данных. Кроме того, с удовольствием послушаем о связанных с этими темами вещах вроде организации CDC, бэкапов или оптимизации файлового хранения.

Популярные СУБД

Расскажите нам о необычном опыте использования, внедрения, масштабирования или администрирования популярных (или не очень) СУБД: PostgreSQL, MongoDB, MySQL, MariaDB, Clickhouse, Couchbase, CouchDB, Greenplum, Elasticsearch, Cassandra, ScyllaDB и т.д.

Нетипичные СУБД

Используете необычную СУБД или классическую СУБД, но по-своему? Было бы интересно послушать про сценарии использования графовых СУБД: ArangoDB, DGraph, Neo4J или XTDB, например, новые объектные СУБД вроде EdgeDB. Может быть, уже есть опыт работы с узкоспециализированным СУБД вроде BigchainDB, Druid, Dolt, TDEngine, kdb+, MLDB? Расскажите!

Хранение в оперативной памяти

Расскажите о своём опыте использования распределённого кэша или in-memory DB: Redis, Hazelcast, Tarantool, Ignite, Ehcache, Aerospike, KeyDB.

Очереди / Брокеры сообщений

Kafka или Pulsar? Или знаете что поинтереснее? С удовольствием послушаем. Кроме того, расскажите об интересном опыте с RabbitMQ, ActiveMQ, NSQ, Nats, Pulsar, RocketMQ, EMQ, KubeMQ, Redanda.

Протоколы обмена данными

В этом блоке интересно всё о переходе на HTTP/3 и о других протоколах вроде GPRC, RSocket, WebSocket и их преимуществах и недостатках. А может быть, вы используете собственный протокол? Расскажите об этом.

DataScience, ML, AI

Придерживаясь традиции CodeFest, ищем самые продвинутые, интересные и зажигательные доклады в секцию Data Science. Нам интересно узнать о том, как работают AI-штуковины сейчас и чем мы будем пользоваться в повседневной жизни через пару лет.

SOTA из первых рук

В индустрии принято reimplementить результаты из научных статей и пытаться их продать. Круто послушать как раз авторов этих статей: настоящих учёных из настоящих лабораторий со свежими материалами.

Уже внедрённые в работу штуки

Разговоры о прекрасном AI будущего это хорошо, но было бы здорово посмотреть на что-то, что уже работает, внедрено и вовсю используется. А ещё лучше, чтобы можно было дать пощупать аудитории это самое:)

Edge devices

Мы уже достигли желаемых точностей в большинстве задач. Теперь вопрос, как перенести их решение с дорогих серверов на мобильные телефоны и встраиваемые устройства. Этот пункт с изяществом объединяет всё от квантования и до AutoML.

Метавёрс

Последнее время на пике популярности. Хорошо бы послушать экспертное мнение о том, как его реализовать-то и на каком этапе сейчас технический прогресс.

Инструментарий

Всё, что поможет быстрее, эффективнее и приятнее двигать Data Science вперёд. MLOps, BigData, подготовка и обработка данных, новости из мира DL фреймворков.

MLOps

Не только про инструменты, но и про процессы непрерывной поставки ценности командами Data Science.

Few-Shot Learning

Как создать полезный продукт, не имея под рукой готового датасета и ресурсов на его сбор. Лайфхаки из маленьких доменных областей и борьба с возникающими проблемами.

Информационная безопасность в DL

Какие вопросы безопасности нужно закрыть, прикручивая новую модельку с Hugging Face к своему сервису. Закладки в DL-моделях и другие опасные моменты при использовании открытых решений.

Цифровые двойники и что с ними делать.

Куча сервисов, которыми люди пользуются каждый день, порождают ещё большую кучу данных: как собирать, обрабатывать и использовать цифровые аватары людей в медицине, маркетинге, безопасности и т.д.

Интерпретация моделей

Как ответить бизнесу на вопрос «почему так?».

DS в экзотических областях

Истории о том, как мы подключили датчики к картошке и на основе детектора аномалий ищем поля, заражённые колорадским жуком.

Frontend

Backend-разработка на JS

Есть ли будущее у конкурентов Node.js, таких как Deno и Bun? Кто уже использует их в продакшене? У вас NPM или вы перешли на альтернативные менеджеры пакетов, такие как yarn, pnpm. Или, может быть, пакетные менеджеры не нужны вовсе и вы перешли на ES Modules?

Serverless

Если у вас есть история про то, как вы построили всю серверную часть на решениях вроде AWS Lambda, то мы бы хотели её услышать! Мы также хотим обсудить возможности и ограничения решений, таких как AWS Lambda, и советы о том, когда и как их лучше использовать.

Real user monitoring (RUM)

Нам интересно узнать, как вы собираете пользовательские метрики и какие решения на их основе принимаете. Может быть, вы уже наступили на все грабли в этой области и хотите поделиться своим опытом с сообществом? Или, может, уже смогли помочь бизнесу, грамотно реагируя на собранную статистику?

Современные подходы к сборке и поставке фронтенда

Какие подходы к сборке фронтенда вы используете? Собираете ли отдельные бандлы под современные и старые браузеры? Может быть, собираете отдельные бандлы для клиента и для сервера? Как вы поставляете свой фронтенд конечному пользователю — у вас фронт это статика, а может быть, у вас SSR или изоморфный/гибридный рендеринг? Какие инструменты вы используете и почему выбрали именно их?

Если у вас есть какой-то другой уникальный опыт, идея или подход, которым вы хотите поделиться с сообществом, то дерзайте и заполняйте Call for Papers — мы будем рады познакомиться!

No-code и low-code

Мы хотим встретить лично людей с опытом построения бизнеса на low-code и no-code решениях. Реально ли использовать подобные инструменты для более комплексных решений, чем лендинги и рекламные сайты. Интересно, кто дошёл до границ применимости этих технологий и как работать с их ограничениями.

PWA

Тема PWA очень активно обсуждалась в сообществе в прошлом, но с тех пор, как нам кажется, немного отошла на второй план. Но технологии вокруг PWA не стоят на месте, и нам интересно, какие возможности там появились и как вы их используете?

Применение AI во фронтенде

Эта тема посвящена исследованию возможностей использования искусственного интеллекта (AI) во фронтенд-разработке. Мы будем обсуждать различные подходы и технологии, которые позволяют интегрировать AI в веб-приложения для улучшения пользовательского опыта и автоматизации рутинных задач. Кстати, описание этого пункта сгенерировано в ChatGPT.

Подходы к обеспечению качества

- QA в фазе Discovery или как можно было начать думать о качестве ещё раньше? Как на самых ранних этапах начинать проектировать качество, а также осознавать его «достижимость» в реальных условиях?
- Как на самом деле обстоят дела с внедрением обеспечения качества на всех этапах? Как реализуются эффективные примеры отказа от выделенной роли тестировщика и/или от выделенного этапа процесса? Как обеспечение качества становится осмысленной общей задачей и какие метрики инженерной производительности помогают решать эту задачу хорошо?
- Shift right. А вы оцениваете удовлетворённость конечного пользователя?

Автоматизация тестирования

- Мы бы с радостью рассмотрели классные реализации нативного автотестирования;
- А также автотестирование групп микросервисов, инфраструктурные проблемы и победы, опыт эффективного контрактного тестирования;
- Всё, что касается ML в автоматизации или ИИ внутри процессов тестирования. Настают ли «тёмные времена» и есть ли шанс, что и тут инженеры могут стать не нужны?

- Автоматизация тестирования в Game dev;
- Лучшие практики анализа эффективности автотестов и любые, действительно, эффективные подходы к оценке тестового покрытия.
- Сложные и неизменно правильные примеры инфраструктуры запуска тестов и сбора результатов;
- Как понять, в какой момент стоит начинать тратить деньги (и какие) на автоматизацию/метрики в тестировании?

Актуальные инженерные вопросы

- CI/CD, если что-то важное относительно задач обеспечения качества ещё не сказано, а вы можете это исправить;
- Вопросы, связанные с тестированием инфраструктуры и обеспечения качества инфраструктуры в целом;
- Тестирование самого ИИ;
- Решения из области нагрузочного тестирования широко применимые или строго специализированные на принесении конкретной пользы;
- Всё важное относительно cybersecurity и соответствующего качества цифровой действительности;

Актуальные примеры спецификации обеспечения качества

- Как уже изменились и продолжают меняться инструменты обеспечения качества и подходы к тестированию за последний год;
- Какие вызовы формируются в области менеджмента обеспечения качества: гибкая адаптация к актуальным потребностям бизнеса, специфика найма и адаптации сотрудников, настройка передачи знаний;
- Волнующие и актуальные вопросы, связанные с системами тест-менеджмента;
- Data quality engineering — хотим обстоятельно поговорить о качестве данных;
- Нам будет интересно рассмотреть предложения, касающиеся обеспечения качества сервисов, связанных с медиа и коммуникациями, со стриминговыми площадками и образовательными продуктами, которые успешно обеспечивают сохранность и доступность своего контента и продукта в целом;
- Как строится качество штук, которыми мы все пользуемся или начинаем пользоваться в реальной жизни: от игровых приставок до беспилотных транспортных решений;
- Будем рады обсудить вопросы, связанные с обеспечением качества доступности цифровой среды;

Развитие ролей и аккумуляция экспертиз

- Динамичная цифровая трансформация формирует спрос на новые гибридные роли. Давайте разбираться, что из этого на самом деле про качество;
- Не забываем, что иногда новое — это хорошо забытое старое. Будем рады поговорить об актуальных примерах разработки в тестировании/SDET и качественном стремлении к T-Shaped.

И, разумеется, если вы делаете что-то потрясающее в области обеспечения качества, о чём мы ещё даже не догадываемся, обязательно сообщайте о себе в Call For Papers. Мы мечтаем с вами познакомиться!

Mobile

SwiftUI/JetPack Compose (ios/android)

- Расскажите про ваш опыт использования в проде.
- Словили много подводных камней и теперь знаете, как из них выбираться,— расскажите.
- История миграции.

Cross Platform (android/ios)

- Необычные кейсы
- Подводные камни
- Необычная архитектура

Архитектура (android/ios)

- Хороший пример использования и сложные кейсы
- Расскажите про реальные цифры эффективности
- Необычный кейс с многомодульностью

ML в мобилке (android/ios)

- Кейс применения
- История успеха
- Реальная польза

Kotlin/Swift (android/ios)

- Много кишочков (компилятор/кодогенерации)
- Крутой кейс при переходе
- История успеха

Автоматизация / безопасность / производительность и т.д. (android/ios)

- CI/CD
- Контроль кода
- Тестирование
- Фреймворки
- Кейс с безопасностью
- Много цифр про производительность

Design

В секции дизайна хотим поговорить про то, как разрабатывают сложные продукты для профессионалов из разных областей. Будут интересны истории про:

- Как дизайн помогает решать самые нетривиальные задачи для инженеров, аналитиков, разработчиков, ученых и специалистов узкой направленности
- Решение дизайнерами задач из нетривиальных сфер — заводы, производства, сельское хозяйство, медицина, администрирование, искусственный интеллект, голосовые ассистенты, боты и другие
- Сходства и различия продуктовой работы b2b, b2e и b2c секторов
- Кейсы, подходы, трудности, лайфхаки и особенности в разработке сложных продуктов и про то, как эти знания помогут в работе над массовыми продуктами

Квартирники

CodeFest — это не только официальная программа, но ещё и народная. Ведь CodeFest — это в первую очередь общение!

Квартирники — формат, который есть только у нас! Квартирники предполагают дискуссию. Здесь нет докладов, зато есть разговоры на темы, которые волнуют участников CodeFest.

Обычно кто-то выступает в роли массовика-затейника, задаёт тему обсуждения, собирает несколько «экспертов», мнение которых ему кажется интересным, и зовёт участников CodeFest присоединиться к обсуждению.

Вот как [это было на CodeFest 12](#)

Квартирники организуют сами участники. Программный комитет никак не модерировать квартирники. Вы присылаете заявку, мы смотрим на актуальность темы в целом для аудитории CodeFest, и если всё ок — ставим вас в расписание. Далее всё в ваших руках: подготовиться к дискуссии, собрать экспертов, раскрыть тему.

Традиционно в этом году часть программы Квартирников сформируем заранее. Если у вас уже есть что рассказать — скорее [отправляйте нам заявку](#).

Податься на Квартирники можно будет и в дни проведения конференции. Принцип действия простой: придумали тему, записались в свободный слот, тема сразу появилась в расписании на сайте. Собрали экспертов, пригласили участников в чате [@cdfst_chat](#) и устроили холивар.

Длительность одного квартирника — 40 минут. Можно быстрее. Дольше нельзя, если это, конечно, не оговорено заранее.

Live Channel

По умолчанию все выступления проходят офлайн, но у нас снова предполагается чисто онлайн-новый канал Live Channel. Вдруг вы захотите выступить онлайн в рамках этого канала.

Live Channel — полностью бесплатный онлайн-канал для всех участников CodeFest.

- Эксклюзивная программа со спикерами, которых нет в офлайне.
- Большая аудитория онлайн- и офлайн-участников.
- Смотреть можно с компьютера или телефона.
- На площадке в офлайне будет организован кинозал, откуда офлайн-участники смогут смотреть, слушать и задавать вопросы спикерам онлайн.

Если вы очень хотите выступить на CodeFest, но не можете приехать к нам в Новосибирск — [отправляйте заявку на онлайн](#)



**Подать заявку
на доклад**

<https://13.codefest.ru/speakers/ru/call-for-papers/>

**Программный
КОМИТЕТ**

program@codefest.ru