CAST 2019

K Keynote N Networking Break R Registration S Session T Tutorial U Unconference/Recreation

W Workshop

AUGUST 12 • MONDAY

7:00am – 7:45am U **Morning Chair Yoga with Dawn (Yoga for Testers)** Starfish

Join Dawn in the morning to learn a few yoga hacks you can take back to the office! Chair Yoga is a gentle, beginner introduction to Yoga. Using your chair or a desk, you can do a pose or two to stretch, get unkninked from sitting in a chair too long, or just mentally ready to go to that meeting.

What to bring: Wear comfortable clothes, bring a water bottle, and best to hold off on drinking coffee or tea until after the session.

Sponsored by PerfTestPlus

8:00am – 9:00am N **Breakfast** Horizon

8:00am – 9:00am R **CAST 2019 Registration** DUNES PREFUNCTION

9:00am – 5:00pm T **TUTORIAL: Just-in-Time Software Testing – Powerful Tools for Fast-Changing Projects and Priorities** Sandcastle

Speakers: Robert Sabourin, P. Eng.

Turbulent development projects experience almost daily requirements changes, user interface modifications, and the continual integration of new functions, features, and technologies. Keep your testing efforts on track while reacting to changing priorities, technologies, and user needs. This highly interactive workshop offers a unique set of tools to help you cope with—and perhaps even flourish in—what may seem to be a totally chaotic environment. Practice dynamic test planning and scheduling, test idea development, bug tracking, reporting, test triage, exploratory testing, and much more.

Be ready for just about anything that can happen in a software testing project such as a complex, customer-facing Mobile, Web, e-commerce or embedded applications. Learn to identify, organize, and prioritize your testing “ideas.” Respond effectively to business, technological and organizational and cultural changes to your testing projects. Create workflows to schedule testing tasks dynamically and adapt the testing focus as priorities change. Decide on purpose what not to test— not just because the clock ran out!

Just-In-Time Testing (JIT) approaches are successfully applied to many types of software projects—commercial off-the-shelf applications, agile and iterative development environments, mission-critical business systems, and just about any application type. Real examples demonstrate how JIT testing either replaces or complements more traditional approaches. Examples are drawn from insurance, banking, telecommunications, medical, and other industries. The course is packed with interactive exercises in which students work together in small groups to apply JIT testing concepts.

Who Should Attend

This course is appropriate for anyone who works in fast-paced development environments, including test engineers, test managers, developers, QA engineers, and all software managers.

9:00am – 5:00pm T **TUTORIAL: The Whole Team Approach to Testing in Continuous Delivery** Sawgrass

Speakers: Lisa Crispin, Ashley Hunsberger

Is your team puzzling over how to feel confident releasing to production frequently with continuous delivery? Delivering reliable and valuable software frequently, at a sustainable pace (to paraphrase Elisabeth Hendrickson), is a worthy goal. DevOps is a hot buzzword, but many teams struggle with how testing fits in, keeps up, and contributes to the DevOps culture.

In this hands-on workshop, participants will have a chance to practice techniques that can help teams feel confident releasing more frequently. You'll learn how your team can use a test suite canvas to discuss what questions each step in your delivery pipeline needs to answer, to understand the value each step provides. You'll work in groups to come up with new experiments to help shorten feedback cycles, make sure all essential types of testing are done continually, and fit testing into the continuous world. You'll learn that there IS a "test" in "DevOps".

Whether your tests take minutes or days, and whether your deploys happen hourly or quarterly, you'll discover benefits. You'll participate in a simulation to visualize your team's current path to production and uncover risks to both your product and your deployment process. No laptops required, just bring your curiosity.

Learning outcomes:

- Continuous delivery concepts at a high level, and the differences between continuous integration and continuous delivery
- Common terminology and a generic question list to engage with pipelines as a practice within your team
- How to use the test suite canvas to design a pipeline that gives your team confidence to release frequently
- Experience in analyzing pipelines from different perspectives to create a layered diagram of feedback loops, risks mitigated, and questions answered
- Ways your team can design experiments to address the many challenges of testing in a continuous world

9:00am – 5:00pm T **TUTORIAL: Visual Validation for Test Automation** Dunes 1/2

Speakers: Angie Jones

Functional test automation is a wonderful way to frequently and expeditiously execute regression testing. However, the test scripts that we write are limited to the few assertions we've considered. Many times, these assertions only cover the tip of the iceberg and account for a small fraction of what a human being would have subconsciously verified.

For example, a test automation script can verify that when adding 2 and 2 via a calculator app, the sum that is returned on screen is 4. But does the 4 appear correctly? Is it upside down? Or sideways? Is it the right color? Are there errors that appear on other areas of the screen? These are all things that the human eye would notice, but an automated regression test would not. The test would continue to pass, even with all of the aforementioned errors. This is where visual validation comes in!

Visual validation is a relatively new concept to add to your test automation toolbox. Applitools, a sophisticated visual validation tool, uses AI to mimic the human eye and brain to verify the look and feel of your application.

In this workshop, you will develop automated UI tests using Java, Selenium WebDriver, JUnit, and Applitools. You'll learn when and where to add visual assertions, how to work with various match levels including ones suitable for dynamic content, and how to evaluate and resolve visual test results.

Existing familiarity with test automation (even if with different tools) will be helpful for this workshop.

10:30am – 11:00am N **Break** Seagrapes

12:00pm – 1:00pm N **Lunch** Horizon

3:00pm – 3:30pm N **Break** Seagrapes

AUGUST 13 • TUESDAY

7:00am – 7:45am U **Morning Beach CrossFit** Walkway to the beach near Tiki bar

Enjoy a beach WOD (work out of the day). Attendees will have a chance to get in a brief, typically CrossFit work out that will require upper body, lower body and core work.
What to bring: Water and goggles

Sponsored by Black Koi Consulting

7:00am – 7:45am U **Morning Chair Yoga with Dawn (Yoga for Testers)** Starfish

Join Dawn in the morning to learn a few yoga hacks you can take back to the office! Chair Yoga is a gentle, beginner introduction to Yoga. Using your chair or a desk, you can do a pose or two to stretch, get unkninked from sitting in a chair too long, or just mentally ready to go to that meeting.

What to bring: Wear comfortable clothes, bring a water bottle, and best to hold off on drinking coffee or tea until after the session.

Sponsored by PerfTestPlus

8:00am – 9:00am N **Breakfast** Horizon

8:00am – 9:00am R **CAST 2019 Registration** DUNES PREFUNCTION

9:00am – 5:00pm T **TUTORIAL: Becoming a Quality Coach** Dunes 1/2

Speakers: Anne-Marie Charrett

Many teams believe and accept that quality is a team responsibility, yet struggle to implement this approach at a practical level. A quality coach provides the necessary assistance and support to enable teams to adopt this strategy. The role of quality coach differs to that of test lead, and requires different capabilities and know how. For one, a quality coach, needs to be able to coach testers and non testers alike, and rather than driving strategy they provide a space for the team to drive the testing strategy.
This one day workshop provides training on the fundamental skills and tools to introduce and implement quality coach initiative.

Content

- Create the Role of Quality Coach (Roles & Responsibilities)
- Quality Attributes Workshop & Measurement
- Quality Assistance for Software Developers
- Coaching Software Testing
- Running Software Testing Workshops

Who should attend
Test Leads who want to become quality coaches

How will it benefit?
Test Leads are being asked to take on the role of Quality Coach where testing is performed by a whole team rather than the speciality role of a tester. This means test leads needs to acquire a different skill sit. This course is designed to help test leads transition to a new role.

What might I learn?

- What is a Quality Coach and how it differs to Test Lead
- Skills required for Quality Coach
- Coaching skills
- Roles and Responsibilities of a Quality Coach
- Creating a Quality Coach Process
- Creating Quality Indicators

9:00am – 5:00pm T **TUTORIAL: Exploratory Testing on Computer Interfaces (APIs)** Sawgrass

Speakers: Maaret Pyhäjärvi

Exploratory Testing is a skilled multidisciplinary style of testing. Many have learned to apply it on user interfaces that naturally speak to testers as their external imagination. Yet with systems of today, it is important we move that skill of smart thinking with external imagination to interfaces hidden from users – public and private APIs. How can you use exploratory testing on something that does not have a GUI?

Let’s shape up our skills of exploring both the functional and parafunctional aspects of a system through its APIs in their operating environments, without forgetting developer experience of having to maintain and troubleshoot these systems. Let’s learn to be intentional with our APIs, instead of being accidental – through delivering relevant, timely feedback.

Intertwining test automation and exploration, we include considerations of the best for today and for the future. For great testing bringing value now as well as when we are not around, we need to be great at *testing* – uncovering relevant information – and *programming* – building maintainable test systems. At the core of all of this is *learning.* What we lack in a set of skills, we can compensate through *collaboration*.

On this course, we are learning these skills hands on testing various interfaces with various tools. It’s a deep dive into applying exploratory testing at computer interface level, not on the tools. The main things you will take out of this session are thinking in terms of coverage, generating versatile ideas on API level, and being aware of opportunity cost to control your own choices as you are testing.

On the side, you will be exposed to ideas such as tools like Postman and Python requests library, to ideas of comparing such as Java Asserts and Approvals, and to approaches such as integration and contract testing. You don’t need to know any of these, but if you do, you’ll still pick up a lot to improve your testing.

Let’s work on code-oriented tests that are not regression oriented but aim at finding new information by means of exploratory testing.

Learning objectives:

- Learn to uncover information on API that a typical automation focus would miss
- Uncover how other people think when testing to learn from your peers
- Learn to identify variables (things you can change) on an API or a method signature
- Learn to prioritize your testing efforts for most relevant feedback fast

9:00am – 5:00pm T **TUTORIAL: What’s Your Problem? – Hands-on Problem Solving Workshop** Sandcastle

Speakers: Fiona Charles

Problem solving is an essential skill for testers—in fact, for anyone in software development. But too often, we jump into trying to develop solutions before we really understand the problem. Even if we’ve identified the problem pretty well, we can get bogged down with too many options, or fall in love with an idea that isn’t a good solution.

Come spend a day working with others to explore and practice techniques to:

- Identify and prioritize the real problem(s) you need to solve
- Evaluate potential solutions
- Apply heuristics to help actually solving problem(s)
- Get unstuck when you have too many options, only one idea, or none at all

We can all get better at solving problems. Let’s have fun learning from each other in this hands-on workshop.

10:30am – 11:00am N **Break** Seagrapes

12:00pm – 1:00pm N **Lunch** Horizon

3:00pm – 3:30pm N **Break** Seagrapes

5:30pm – 7:00pm U **Tester "Game Night" - Pizza & Open Bar** Horizon

Join us for a hosted pizza, bar, and game night.

Sponsored by PerfTestPlus

K Keynote N Networking Break R Registration S Session T Tutorial U Unconference/Recreation

W Workshop

AUGUST 14 • WEDNESDAY

7:00am – 7:45am U **Morning Beach Yoga with Pau** Walkway to the beach near Tiki bar

Rise and shine with a Vinyasa yoga class for all levels. This class links movement and breath to attain balance in the mind and body. You will stretch your body, strengthen your muscles and calm your mind with a sweet meditation at the end, feeling renewed, relaxed and recharged. No previous experience necessary.

What to bring: Water, comfortable clothes and yoga mats or towel.

Paula D’Elía
Yoga teacher and Health Coach
Www.pausecrets.com
IG @pau_secrets

Sponsored by PerfTestPlus

7:00am – 7:45am U **Morning Chair Yoga with Dawn (Yoga for Testers)** Starfish

Join Dawn in the morning to learn a few yoga hacks you can take back to the office! Chair Yoga is a gentle, beginner introduction to Yoga. Using your chair or a desk, you can do a pose or two to stretch, get unkninked from sitting in a chair too long, or just mentally ready to go to that meeting.

What to bring: Wear comfortable clothes, bring a water bottle, and best to hold off on drinking coffee or tea until after the session.

Sponsored by PerfTestPlus

7:30am – 8:30am U **Lean Coffee** Horizon

Speakers: Lisa Crispin

8:00am – 9:00am N **Breakfast** Horizon

8:00am – 9:00am R **CAST 2019 Registration** DUNES PREFUNCTION

9:00am – 9:15am S **CAST Welcome** Sea Oats

9:15am –
10:15am

K **Observability and Complex Systems: What Got You Here Won’t Get You There** Sea Oats

Speakers: Charity Majors

Distributed systems, microservices, containers and schedulers, polyglot persistence … modern infrastructure is ever more fluid and dynamic, chaotic and transient. Likewise, individual engineering roles can no longer be broken down neatly into software engineers (who write the code) and ops engineers (who deploy the code (and buffer the consequences)). Many teams have already sailed past an event horizon of complexity and found that their old tools and processes — and org charts! — no longer work for them.

But why, exactly?

- What was the matter with traditional metrics and logs?
- Why are they failing to keep pace with modern systems? What else is out there?
- Isn’t observability just a marketing term that means monitoring?
- What else do we have to look forward to that will fail us?

In this talk we’ll cover the technical and cultural differences between monitoring and observability. Operations is now part of every engineer’s mandate, just like testing became part of every engineer’s mandate a decade ago. What does this means for specialist practitioners like test and ops engineers? What model of software ownership should we be working towards in our organizations, and why? Is it the same for everyone? And how does chaos engineering fit into this whole mess: is it a must-have or a nice-to-have, and why?

10:15am –
10:45am

N **Break** Seagrapes

10:45am –
11:45am

S **Being a Tester After Trying Almost Everything Else** Sawgrass

Speakers: João Proença

Ten years ago, I was a tester. Until I became so disgruntled with my work that I didn't want to do it anymore. Ever again. So, I started doing a lot of other things instead. I was a developer, a support engineer, I led an Ops team, and even worked In Marketing! And finally, I returned to the quality world.

Funnily enough, working on so many different roles allowed me to become a better engineer and tester. So I'd like to share with you how trying out new roles made me develop new skills that were also useful for my job as a tester. It also improved the way I interacted with different people and made me look at things differently. For example, do you know what's usually the most effective way to diagnose a bug submitted by a customer? Or can you guess what I struggled with the most as an engineer finding his place in a marketing department? I'll answer these questions and more!

Takeaways

- Understand how context can make someone disgruntled with testing and not the Job itself.
- Trying out new roles can make you a better professional in ways you never imagined.
- Knowing how a developer, customer support professional or others think allows you to better interact with people in these roles.

10:45am –
11:45am

S

It's Not Rocket Science, It's Far Trickier

Dunes 1/2

Speakers: Nicola Sedgwick

Popular culture leads us to believe that rocket science and brain surgery are the most complicated things out there. "It's not rocket science" is uttered when the business describe the new feature they want, or engineers explain the technical architecture or marketing want details of new features for the website announcement. This ex-rocket scientist thinks that most of what we do is far more complicated that rocketry (though admits she's completely unqualified to comment on the brain surgery comparison) and has been known to retort "rocket science is ok, it's recovery that's tricky".

In a 40 minute talk we will cover a number of known live rocketry failures due to undiscovered bugs, major test failures during construction, as well as missions enhanced by successful collaboration. These examples will be linked back to practices that the testing industry may replicate or may rail against, for instance full end-to-end regression tests after integration of individual components. The aim is for attendees to learn that taking inspiration from other industries can enhance their craft and increase the diversity of their test ideas.

10:45am –
11:45am

S

My Love Affair With Testing (and you can have one too!)

Sea Oats

Speakers: Jess Ingrasselino, Ed.D

Have you ever wondered how you can tie your life interests into your work? Do you feel like your many interests fit together and inform one another, but struggle to articulate how? Do you love to learn, and want to understand how your love for learning can joyously infect your entire life, including your testing? Jess will share pedagogies of learning (Zone of Proximal Development, Scaffolding, Metacognition, and Flow), and then do a live demonstration weaving testing, music, and other disciplines together in a fascinating talk where she lays her skills on the line to show you how you can tap into your passion for anything and turn it toward testing.

11:45am –
2:00pm

N

Lunch-CASTiesta

Horizon

2:00pm – 3:00pm S **Creating Test Stability to Create Continuous Delivery** Dunes 1/2

Speakers: Trisha Chetani

Our agile teams struggled to continuously merge local code changes to the master repository. In our company, our automated test were taking 30 minutes of execution time and the occurrence of flaky tests was just multiplying this time and reducing the confidence in the results. This slowed our agile teams down tremendously to the point where they couldn't deliver continuously with automated pipelines. Ensuring high quality for each iteration became a challenge for our agile teams as they had to wait for my analysis of the failing tests. We had created automated checks to make our testing more efficient, but instead they were slowing us down and causing bottlenecks - I was a bottleneck.

Our automated tests were maintained by various team members and some of them were not following leading practices - which led to flaky results due to such things as - concurrency relying on non-deterministic behaviour, caching, dynamic content and many other factors. I needed to find out the root causes. I started my investigation and found the common issues were because of environment, locators, coding practices and a lack of knowledge sharing and code reviews. I improved our locators, coding practices, debugging and simultaneously the developers helped us to fix all environment issue and slow page load times, it was a long journey but it was definitely worth it.

Currently our automated test suites takes just 5 minutes to run in our CI environment - a huge improvement! Now my team members are more encouraged to see why a test is failing because the majority of failures now are genuine, and are providing valuable information. As a team, we get to spend more time testing and creating new automated tests, which has resulted in finding more bugs. We've also put things in places to avoid ending up in a flaky nightmare again, such as a Wiki Page of leading practices.I'm no longer a bottleneck, the team works together to maintain fast valuable automated tests, and I want to share the insights from that journey with you.

After this session, you'll learn techniques to:
1. Treat test code as production code and test environments as production environments
2. Improve the stability of your tests which in turn improves your continuous delivery pipeline
3. Make tests faster to enable fast feedback

2:00pm – 3:00pm S **Quality is a Team Responsibility** Sea Oats

Speakers: Anne-Marie Charrett

We are told that when quality is owned collectively, as opposed to being gated by a tester it can improve overall quality. Teams begin to contribute to quality through testability, identifying design flaws in stories, and testing earlier in the delivery process.

But this only scratches the surface of what can be achieved.

If we think of software delivery as a journey to achieving desired business outcomes, quality is the GPS that shows us where we are and how far we have to go.

This talk looks at some ways to better understand quality, what it means to you, your team and business partners. It looks at ways to frame quality in terms of business outcomes to help us keep on the right track. It will show ways to visualise quality so everyone can see where we are on our journey.

Do I believe testers have a place in this future? Absolutely! In fact, I think the role of a tester is needed more now than ever but perhaps not in a way we have traditionally seen our roles.

A must for anyone moving to contemporary engineering approaches.

2:00pm – 4:15pm W **Thinking outside the box: building awareness of unconscious biases for more effective testing**

Speakers: Lisa Crispin, Rachel Kibler Sawgrass

Finding bugs requires detective work. You look for evidence, you interrogate witnesses - and some of those might not be telling the truth! You need lateral thinking to find your perpetrator.

As testers we often hear that the capability to think outside the box is an essential skill to have. In fact, this skill is great to have even if you're not a tester.

Thinking outside the box permits you to generate new and innovative ideas, to find bugs that nobody could think about, it can also help you find completely new ways to solve problems.

We all have unconscious biases that can limit our observational skills and creativity. Rachel and Lisa will introduce lateral vs. vertical thinking, and explain how our brain's "wiring" can make thinking outside the box be difficult.

In this playful workshop, you will discover several common cognitive biases that can work against us and get in the way of effective testing. You'll practice some games that promote lateral thinking, in a safe and fun environment. You'll leave the workshop with new skills and guidance on how to keep building up your lateral thinking "muscles".

3:00pm – 3:15pm N **Break** Seagrapes

3:15pm – 4:15pm S **5 Levels of API Test Automation** Sea Oats

Speakers: Shekhar Ramphal

In my context we run a micro service architecture with a number (300+) of api endpoints both synchronous and asynchronous. Testing these in a shared environment with cross dependencies is both challenging and very neccessary to make sure this distributed monolith operates correctly. Traditionly we would test by invoked an endpoint with the relevant query params or payload and then assert the response code or body for valid data / type definitions. This proved to be more and challenging as the push for CI and having common data sources meant dependencies would go up and down per deployment which meant flakey tests.

I will demonstrate how we leveraged of newer technologies and split our api testing into 5 levels to increase our overall confidence. The levels are: (ignoring developer focused unit and unit integration tests)

1. Mocked black box testing - where you start up an api (docker image) identical version to the one that would go to PROD but mock out all its surrounding dependencies. This gives you freedom for any known data permutations and one can simulate network or failure states of those dependencies.
2. Temp namespaced api in your ci environment - here you start up ur api as it would in a normal integrated env but it's in a temp space that can be completed destroyed if tests fail… never gets to the deploy stage and no need to roll back if errors occur, use kubernetes and ci config to orchestrate these tests. The tests focus is to check 80-20 functionality and confirm that the api will meet all the acceptance criteria.
3. Post deployment tests - usually called smoke testing to verify that an api is up and critical functionality is working in a fully integrated environment.

We should be happy by now right? Fairly happy that api does what it says on the box… but…

1. Environment stability tests - tests tha run every few min in an integrated env and makes sure all services are highly available given all the deployments that have occurred. Use gitlab to control the scheduling.
2. Data explorer tests - these are tests that run periodically but use some randomisation to either generate or extract random data with which to invoke the api with. These sort of tests are crucial for finding those edge cases that are usually missed. Sometimes of low occurrence but generally high risk issues. I wrote a custom data extractor that runs against our DBs to find strange data sets to use as tests data.

I would like to elaborate and demonstrate these layers and execution and how this has changes the way we test and look at APIs. Would also touch on the tooling we use to achieve this and the pros/cons of using this approach.

3:15pm – 4:15pm S **Making the Grade - Testing Undergraduate Software Engineers** Dunes 1/2

Speakers: Robert Sabourin, P. Eng.

As an Adjunct Professor of Software Engineering at McGill University, Robert Sabourin does a lot of grading and scoring of students using a variety of academic testing processes. Over the past twenty years Robert has evolved many different mechanisms to test students extending the notion of open book to open university and beyond. Rob's students are generally expert programmers with strong math and engineering skills but with little knowledge of the process, methods and practices of professional software engineering. Rob will share his experience with several entertaining stories. He will walk through some rubrics and grading schemes. Rob will explain "testing" conundrums he has faced in the continuous battle between a bunch of intelligent students trying to do the minimal amount of work to get the highest possible grade. Rob hopes to inspire an interesting open season discussion to see if there are lessons, we can draw from grading undergraduates. Rob promises an energetic fun filled session about his adventures warding off the paper chase and focusing students on learning the right things well.

4:15pm – 4:30pm N **Break** Seagrapes

4:30pm – 5:30pm S **Building Automation Engineers From Scratch** Sea Oats

Speakers: Jenny Bramble

Creating automation engineers from manual testers is hard. Even if your testers are willing, they have a lot of hurdles to get over to feel like the same kind of subject matter experts. I've been there done that, currently going through that and have advice to give for testers and their management.

Creating automation engineers from manual testers is hard. Even if your testers are willing, they have a lot of hurdles to get over to feel like the same kind of subject matter experts in automation as they are in manual testing.

As a career-long manual tester making the leap to automation, Jenny Bramble has experience to explain frustrations and provide solutions. She will discuss managing the expectations of your testers and their managers (what's the time frame? Why isn't this working?), techniques for teaching (such as games! Pair/mob programming! Software fundamentals!), and how to know when your testers have made it (what should manual testers be aiming for when they start?). You'll walk away from the talk feeling empowered to create a plan to build your automation engineers from scratch!

You'll walk away from this talk with a powerful new set of tools in your toolbox:

- The basic framework your manual testers need to be successful—including how to determine where the gaps in knowledge are and filling them.
- Advice on managing the expectations of your testers and management from time constraints to what success looks like.
- Several methods for teaching framed around a case study of a team that built itself up from the inside out and is running a successful automation suite.
- Facing and overcoming other challenges such as ability and perceived ability, resources, time, tooling, and how to get your team excited for a new chapter in their professional development.

4:30pm – 5:30pm S **Coaching Your Team to Test** Dunes 1/2

Speakers: Ali Hill

As the sole tester on a team that's moving towards continuous delivery and building a DevOps culture, how can your team release frequently, and with confidence?

Within my Agile team, the testing activity had become the bottleneck. The testing 'To Do' cards on the team's Kanban board were piling up. As the sole test specialist within my team I felt as if I was preventing us from being able to deploy code to our live environment. Frustrated, we got together as a team to discuss how we could fix this problem.

Our solution? We decided to share my exploratory and automated testing knowledge with the team. My role as the test specialist evolved into a coaching role, which made me feel both excited and nervous. We found ways to test throughout the development process. We learned to design test plans and discuss technical challenges together. We collaborated on the testing effort.

In this talk I'm going to share how my team removed the testing bottleneck, built in quality to our product and started to become true cross-functional team members by increasing collaboration. If you face similar challenges with your own team, you can try similar experiments.

7:00pm – 11:00pm U **Welcome Reception - Kennedy Space Center** Kennedy Space Center

Thanks to PerTestPlus for hosting this year's welcome reception at the amazing Kenny Space Center. Just like last year, this is going to be a BLAST! (pun intended)

K Keynote N Networking Break R Registration S Session T Tutorial U Unconference/Recreation

W Workshop

AUGUST 15 • THURSDAY

7:00am – 7:45am U **Morning Beach Yoga with Pau** Walkway to the beach near Tiki bar

Rise and shine with a Vinyasa yoga class for all levels. This class links movement and breath to attain balance in the mind and body. You will stretch your body, strengthen your muscles and calm your mind with a sweet meditation at the end, feeling renewed , relaxed and recharged. No previous experience necessary

What to bring: Water, comfortable clothes and yoga mats or towel.

Sponsored by PerfTestPlus

7:00am – 7:45am U **Morning Chair Yoga with Dawn (Yoga for Testers)** Starfish

Join Dawn in the morning to learn a few yoga hacks you can take back to the office! Chair Yoga is a gentle, beginner introduction to Yoga. Using your chair or a desk, you can do a pose or two to stretch, get unkninked from sitting in a chair too long, or just mentally ready to go to that meeting.

What to bring: Wear comfortable clothes, bring a water bottle, and best to hold off on drinking coffee or tea until after the session.

Sponsored by PerfTestPlus

7:30am – 8:30am U **Lean Coffee**

Speakers: Lisa Crispin

8:00am – 9:00am N **Breakfast** Horizon

8:00am – 9:00am R **CAST 2019 Registration** DUNES PREFUNCTION

9:00am – 9:15am S **Opening** Sea Oats

9:15am – 10:15am K **Your Everyday Tester** Sea Oats

Speakers: Ash Coleman

Inspiration comes from anywhere and everywhere. As we navigate through life, we do so in a way that test the limits of our abilities. Questions I have asked myself in the past are, “Can I make it to work on time if I leave now?”, “What happens if I use baking soda instead of baking powder?”, and on days when I’m feeling especially strong, “Can I deadlift 200lbs?”. All of these questions require exploration to get to a large degree of certainty. Opportunities in my daily life have inspired me to ask these questions, however it is only when I put them to the test that I get any answers.

The same goes for testing! Items within my everyday inspire me as a tester. Seeing software for its possibilities, and testing it for its limits help me define degrees of certainty. If we curate software testing with as much curiosity as we explore this world we live in, we will find that a lot more of what we do in real life inspires possibilities when exploring software.

10:15am – 10:30am N **Break** Seagrapes

10:30am –
11:30am

S

Hello World – How I started in AI/ML and how you can too!!!

Sawgrass

Speakers: Umang Nahata

Artificial Intelligence and Machine Learning are making inroads into every conversation. Many people think that AI/ML is a new kid on the block. Surprisingly, it is a blast from the past which seems to be advancing at a rapid pace and will impact our lives significantly.

When I heard about its resurgence, I was curious to learn more about it. I had constraints: close-to-zero AI/ML experience, no data-scientist skills, and little infrastructure at my disposal. I wanted to dip my toes in AI/ML gently and practically, without getting overwhelmed or bored to death by reading obscure and dry books on this subject. I wanted to learn by doing rather than reading.

If you're in a similar situation with regards to getting started in AI/ML, this session will provide you with necessary information and practical guidance. I'll share my experience on how I gently jump-started my journey into AI/ML using Azure ML Studio and how you can too. Then we'll build/tweak/demo an AI/ML example and create a callable API endpoint for us to test and explore.

Furthermore, I'll point you to resources, tips and tricks, which I've found valuable. In the end, you will learn AI/ML fundamentals and how to jump-start your AI/ML journey practically.

10:30am –
11:30am

S

Test Ideation: What Writing Taught Me About Testing

Sea Oats

Speakers: Paul Merrill

Does the creative mind belong in testing? Can drills from writing add anything to our craft as testers? If so, where do they fit in?

Long before my testing journey began, and shortly before studying computer science I was 6 credits from a degree in creative writing. Now, I use these skills daily.

What if testing is both an analytical activity AND a creative one? After meeting and working with thousands of testers and test automation engineers, Paul Merrill has noticed a common struggle for testers is creating tests. With numerous teams, Paul has found several of the drills from creative writing programs can help with directing the mind in testing.

Join Paul for this mini-workshop. Bring a readiness to scribble at a whiteboard or privately in your notes, and an open mind. Learn from those around you and be ready to share your learnings as they come in this fun set of exercises you can take back you your team next week for better testing!

10:30am – 11:30am S

Testing Satellites

Dunes 1/2

Speakers: Vladimir Glavac

Systems designed to be operated in space have a few unique problems to must be addressed.
First, they cost a large amount of money. A “black box”, or in other words, a unit that performs a specific task and is part of a larger system, can range up to several million dollars for one. Considering that redundancy is used to increase the reliability of space systems, these costs can easily double. A small communications satellite, which includes many units, runs into the tens of millions of dollars. A more complicated satellite, such a radar imaging satellite can cost a few hundred million dollars. A constellation of satellites, such as those proposed for worldwide communications are in the range of a few billion dollars. This is just the cost of the hardware that is in space. Hidden costs include development and qualification, launch costs, and operation costs. Worse yet, you cannot go up there and fix it if things break.
These reasons alone is enough to justify a rigorous design, build and test philosophy. This lecture will expand on these ideas, but concentrate on the testing aspects. Verification and testing is different for manned systems compared to unmanned systems. Human life is precious, so manned systems undergo more stringent testing. This lecture will concentrate on unmanned systems, and how they are verified and tested. Specifying how one will verify and test starts from the top down. However, during the build process, testing is done from the bottom up. What is being tested and verified varies with each level of integration. These differences will be explained. Specific examples of testing and verification at the different levels will be shown.

11:30am – 12:45pm N

Lunch

Horizon

12:45pm – 1:45pm S

"Git hook[ed]” on images & up your documentation game

Sea Oats

Speakers: Veronica Hanus

Can you remember the difference between two hex color values? Me neither!
Entering visual representations of recently-changed elements into version control makes review of past changes easier & speeds acclimation to a new web project, especially for visual learners. Surprisingly, methods for including images in your version control aren't standardized and are rarely used outside of large companies, and the rest of us are left checking out every major commit and viewing changes locally! Join me for a review of methods currently in use and discuss the benefits and drawbacks of each. The audience will learn from a survey of tools used by both designers and web developers, what methods are most appropriate for individual projects, & how these methods differ from those used at some of the largest companies (Google, eBay, etc.). Finding a method to track changes in your visual elements will save our future contributors (and future selves!) the pain of having to distinguish #2dc651 (lime green) from #34a34e (darker(!) lime green) and ultimately make our commit histories cleaner and our repos easier to navigate in ways that many of us have never imagined!

12:45pm – 1:45pm S

Lightning Talks

Sawgrass

12:45pm – 1:45pm S

The Origins of Context Driven Testing Abstract

Dunes 1/2

Speakers: Doug Hoffman

How did Context Driven Testing (CDT) and AST come about? Was it one or two people, or a committee effort? Who were the principal people who articulated the ideas? The answers provide an interesting perspective on how CDT evolved. Doug does not claim to be the father of CDT, but was an active participant and observer who recorded many of the activities that led to its creation. The session is a retrospective of the people and experiences that brought us here.

2:00pm – 3:00pm S **Why is There a Marble in Your Nose** Sea Oats

Speakers: Angela Riggs

The first time I asked a student “Why is there a marble in your nose?”, it was a learning experience for me. That initial “why” led me down a path of questions that I hadn’t known I needed to ask - and it revealed that the real challenge wasn’t actually the stuck marble at all! The real challenge I needed to solve was that the student had outgrown naps and needed something to do while his classmates were sleeping.

Software testers come from a wide variety of backgrounds, which often influences how we carry out our role as testers. For instance, the skills involved in helping children transition from playtime to naptime enabled me to lead a rollout of process and workflow changes in our engineering department. My experience teaching math to 5th-grade students with an “I do; we do; you do” approach translates pretty directly to helping engineers learn a new testing framework. And of course, I learned the importance of asking “why”, which has made me a better advocate for users and engineering teams. Why did we make that design choice? Why should we iterate this process? Why isn’t this tool working out for us?

Every experience matters. We all bring a variety of expertise and lessons learned from previous roles or industries - and this is a good thing! It allows us to think outside the box and challenge the status quo; to include perspectives that would otherwise be lost or overlooked. Having a wider breadth of experience and knowledge makes us better testers, and I hope that my talk inspires you to think about how your prior jobs have impacted and improved the way you work in software testing as well.

2:00pm – 4:15pm W **A Testers Guide to the Illusions of Unit Testing** Dunes 1/2

Speakers: Ash Winter

One area that testers might be able to enhance their contributions to software development teams is how we perceive and contribute to unit testing. Being able to influence this type of testing in a positive manner is a skill that testers will need to get to grips with, as more companies start to embrace a model of lone testers in cross functional teams. The shift of focus from primarily the testing that testers do, to the testing that the team does, is a key shift in thinking and behaviour.

To facilitate this shift, I believe testers busting their own illusions about this aspect of building something good would bring us much closer to developers and help us realise what other layers of testing can cover most effectively. The last point is pertinent here, as knowing and guiding unit testing brings the role of integration, acceptance and exploratory testing into sharp focus.

This is a topic that has always intrigued me, having predominantly worked as a single tester on a team for the last five or so years. I reached out to the community with the question “What do testers believe about unit testing?” and received a lot of engagement. The good users of Twitter added another 50 or so illusions that testers might have about this layer of testing. I figured that based on that level of engagement, maybe this would make an interesting talk! It wasn’t only testers who responded too, suggesting that there might be some shared illusions about unit testing that are cross disciplinary.

The growing list delighted me, so I wrote a blog with the raw list, it can be found here
http://testingisbelieving.blogspot.co.uk/2017/12/testers-guide-to-myths-of-unit-testing.html

The list alone is interesting but now I would like to share my analysis of it with you, focusing on:

- Recurring themes within the list and how to address them as a tester or developer.
- Particular illusions to look out for with examples from my recent past.
- A guide for developers to engage with testers on unit testing, and testers with developers.

2:00pm – 4:15pm W **Don't Take it Personally** Sawgrass

Speakers: Bailey Hanna

Receiving feedback can be tough. It can be hard to remember that it is meant to help improve work going forward, not to point out current flaws. It can be incredibly easy to take feedback or comments personally in the workplace, but what is the impact when we do so?

When we personalize situations we tend to lose sight of the bigger picture. It becomes easier to focus in on minute details and not look at the overall context in which the feedback is being given. The impacts of this lower level of focus can result in wasted time from; chasing the wrong issues, laying blame, making up excuses, refusing to ask for help, and ultimately avoiding discussions around the root cause and ways to improve.

This workshop will begin by drawing on experiences and examples of situations such as; testing debriefs (tester to tester interactions), bug discovery (tester to developer interactions), and inter team projects (team to team interactions) and discuss tactics for each on staying objective and productive. Once we've learned some of these tactics we will dive in with some hands on demonstrations before we break off into groups to give them a try on our own; from both sides of the conversation! When we look at feedback for what it truly is - a way to improve - we can build better relationships between communities and teams to make them stronger as a result.

3:00pm – 3:15pm N **Break** Seagrapes

3:15pm – 4:15pm S **Building Deep Thinking Tools for Exploratory Testers** Sea Oats

Speakers: Pradeep Soundararajan (IN)

I have been a functional exploratory tester. I was motivated to move out of exploratory testing and become that cool kid doing automation. Thankfully, someone pulled me aside and told me - I am more suited to be a functional exploratory tester and that I am business savvy. I didn't shy away from code. I worked closely with developers and my testing approach involved reading code (didn't write any) and brought in value to developers and to the business. I grew up as a tester being coached by experts, reading blogs from experts and learning that automation will help exploratory testers do more.

In my wait - I found very little effort that has gone in direction. I partnered with developers to be building some products.

I failed multiple times and here are my failed attempts

1. Tool for Social Media Driven Testing for Testers
2. Tool for mapping the heuristics and oracles to test ideas
3. A checklist tool for testing mobile apps and scoring on quality for start-ups
4. Testing Depth Dashboards

I would like to share

1. The thinking behind building these tools and their value
2. How purely focusing only on automation is taking away the possibility of building tools
3. How I would love the community to start building tools
4. How I can help the audience (without commercial interest) and how they can help me

I would like to tell my own story and how I (actually stealing credit from my team who build it) came to build tools

1. What I did different from other manual testers?
2. What problems concern testing space according to me
3. The problems I picked and the way I tried solving them
4. The failures and analysis
5. The existing opportunity for people to build value different from automation frameworks and scripts
6. The vision of open sourcing non code deep thinking tools for testing

4:20pm – 4:40pm S **Closing** Sea Oats