

# STATE DEVELOPMENT FRAMEWORK

October 6, 2017 PMI-SVC (Sacramento Valley Chapter) Meeting

Al R. Pangelinan, MBA, PMP, PMI-ACP, CSM, CSD, CSPO, MOS

# AGENDA

- Introductions
- Waterfall Model vs. Agile Software Development
- Rapid Application Development
- State Development Framework
- Concerns with SDF

# Introductions

GETTING TO KNOW THE AUDIENCE

# AL R. PANGELINAN

- Project Oversight Corporation
- Information Integration Innovation and Associates, Inc.
- Certifications:
  - Certified ScrumMaster (CSM)
  - Certified Scrum Developer (CSD)
  - Certified Scrum Product Owner (CSPO)
  - Project Management Professional (PMP)

# AL R. PANGELINAN

- Involved with enterprise projects utilizing Agile or Iterative Development:
  - Intel Corporation Web Invoice Project (Programmer Analyst/Project Manager 1999-2000)
  - Secretary of State VoteCal Project (IV&V Consultant 2007-2017)
  - Department of Housing and Community Development (HCD) CASAS Project (IV&V 2015-2016)
  - Department of Food and Agriculture CalCannabis Project (IV&V Consultant 2016-2017)
  - Bureau of Cannabis Control Cannabis Licensing, Enforcement and Reporting (CLEaR) Project (PM Support Consultant 2017-Present)

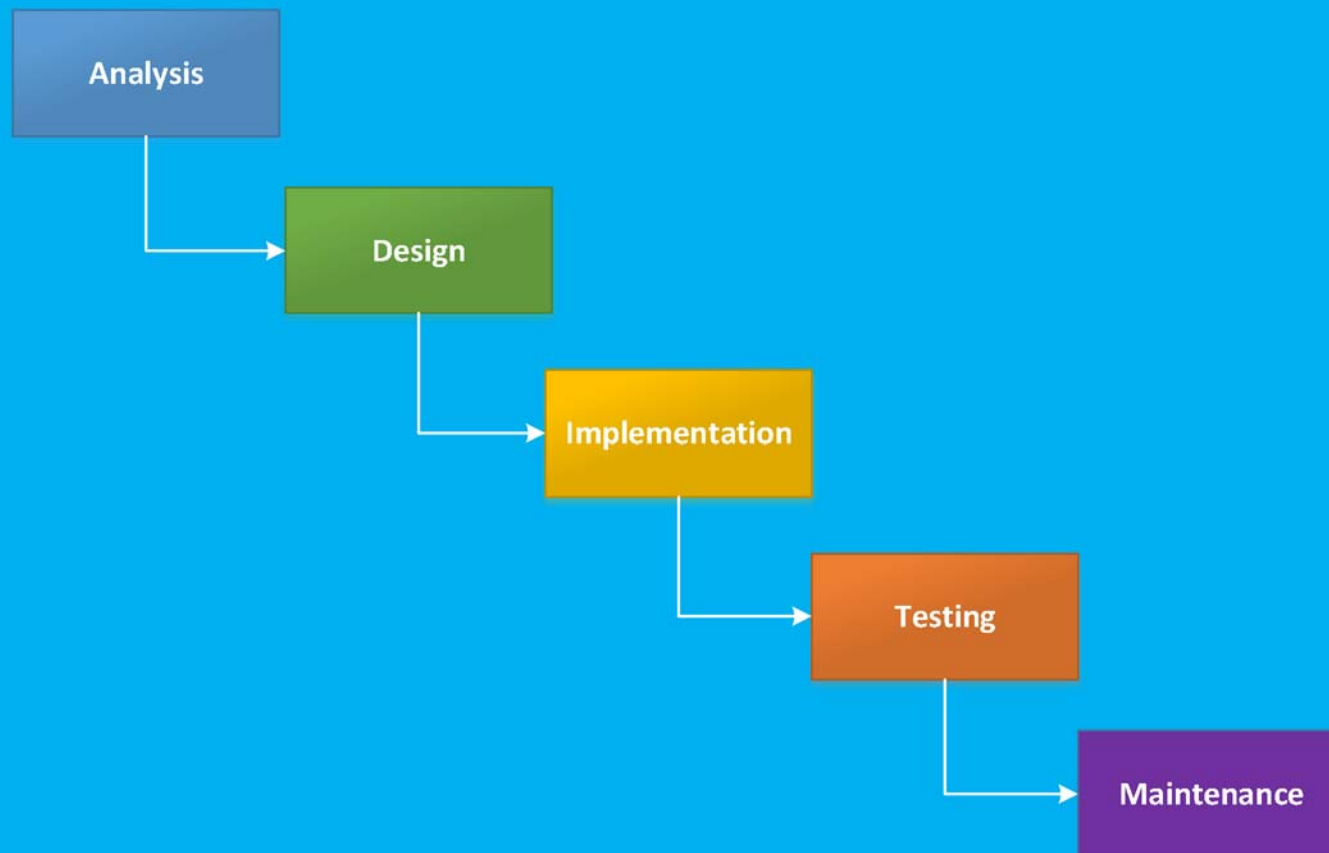
# AUDIENCE

- How many of you are State employees?
- How many of you are consultants with the State?
- How many of you are familiar with CDT CA-PMF?
- How many of you are familiar with CDT SIMM 19 PAL?
- How many of you are familiar with the Agile Manifesto?
- How many of you are familiar with the Agile Software Development?

# Waterfall Model vs. Agile Software Development

TRADITIONAL DEVELOPMENT LIFECYCLE VS. THE EVER-  
CHANGING ITERATIVE DEVELOPMENT LIFECYCLE

# WATERFALL MODEL





# WATERFALL MODEL

## Advantages

- This model is simple and easy to understand and use.
- It is easy to manage, since each phase has specific deliverables and a review process.
- In this model, phases are processed and completed one at a time, because phases do not overlap.
- Waterfall model works well with requirements that are very well understood.

## Disadvantages

- It does not respond well to changing requirements.
- Testing occurs too late in the model.
- It contains high amounts of risk and uncertainty.
- Technology must be known.
- It involves less customer interaction.

# INTRODUCTION TO AGILE

## What is Agile?

The ability to create and respond to change in order to succeed in an uncertain and turbulent environment.

## What is Agile Software Development?

Agile Software Development is an umbrella term for a set of methods and practices based on the values and principles expressed in the Agile Manifesto.

# AGILE MANIFESTO

- **Individuals and interactions** over processes and tools
- **Working software** over comprehensive documentation
- **Customer collaboration** over contract negotiation
- **Responding to change** over following a plan

# AGILE MANIFESTO

- 12 Principles

1. Customer satisfaction by early and continuous delivery of valuable software
2. **Welcome changing requirements, even in late development**
3. **Working software is delivered frequently (weeks rather than months)**
4. **Close, daily cooperation between business people and developers**
5. **Projects are built around motivated individuals, who should be trusted**
6. **Face-to-face conversation is the best form of communication (co-location)**
7. Working software is the principal measure of progress
8. Sustainable development, able to maintain a constant pace
9. Continuous attention to technical excellence and good design
10. Simplicity—the art of maximizing the amount of work not done—is essential
11. **Best architectures, requirements, and designs emerge from self-organizing teams**
12. Regularly, the team reflects on how to become more effective, and adjusts accordingly

# AGILE SOFTWARE DEV ADVANTAGES

- It is more flexibility than the Waterfall Model.
- Iterations provide immediate feedback.
- It gives the ability to respond to changing requirements.
- It involves fewer defects in the final product.
- It involves constant team involvement and collaboration.
- It promotes continuous improvement.

# AGILE SOFTWARE DEV DISADVANTAGES

- It is not ideal for organizations where all aspects of work must be formally planned and documented.
- Immediate feedback can easily result in Scope Creep.
- It is not ideal if communication and decision making process requires multiple inputs and a complex decision process.
- Time and efforts will be continually required from program (business) resources.
- Team members must be highly skilled and cross skilled in competencies as core teams are small.

# AGILE SOFTWARE DEVELOPMENT

- The most popular Agile Software Development Methodologies and Frameworks include the following:
  - Adaptive Software Development (ASD)
  - Agile Unified Process (AUP)
  - Crystal Clear Methods
  - Dynamic Systems Development Method (DSDM)
  - Extreme Programming (XP)
  - Feature-Driven Development (FDD)
  - Kanban
  - **Rapid Application Development (RAD)**
  - **Scrum**
  - Scrumban

# Rapid Application Development (RAD)

EXTRACTING THE RAD ASPECTS FOR STATE USE



# RAPID APPLICATION DEVELOPMENT

- RAD and Scrum share many traits, such as the following:
  - Iterative, Timeboxed Development
  - Customer Engagement
  - Small Teams
  - Adapting to Changing Requirements
  - Quality Assurance
  - Use of Tools

# RAD KEY IDEAS

- Prototyping
- Code Generation
- **Metrics**
- Visualization Tools
- **Best Practices**
- Shared, Usable Domain Models
- **Identifying and Managing Risk**
- Productivity-tool Strategy
- **Joint Application Development**
- Accurate, Rather than Precise, Estimation
- Process Support Tools

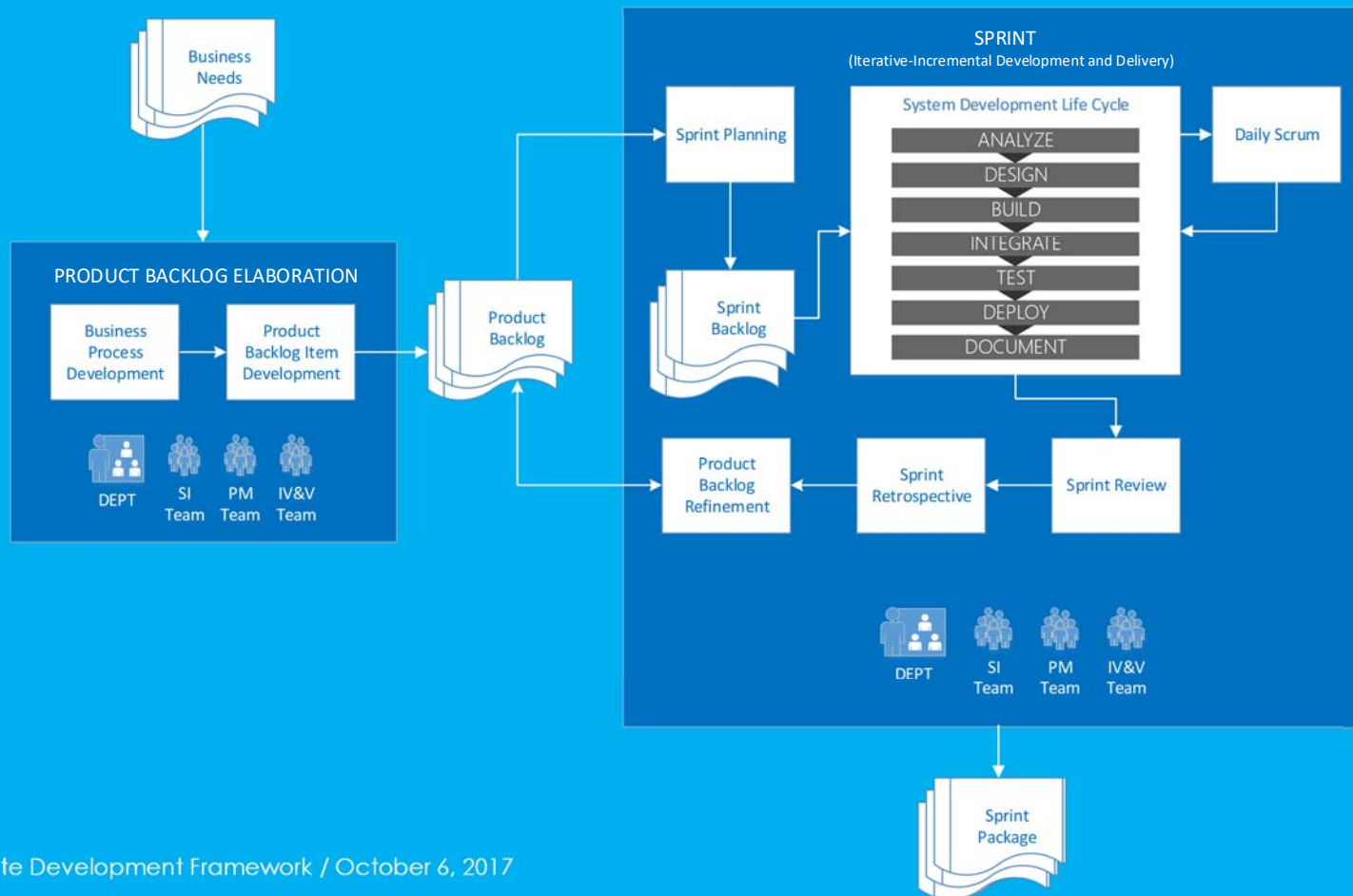
# State Development Framework (SDF)

ACQUIRING THE BENEFITS OF RAPID APPLICATION  
DEVELOPMENT AND SCRUM

# STATE DEVELOPMENT FRAMEWORK (SDF)

**SDF = RAD + Scrum**

# STATE DEVELOPMENT FRAMEWORK (SDF)



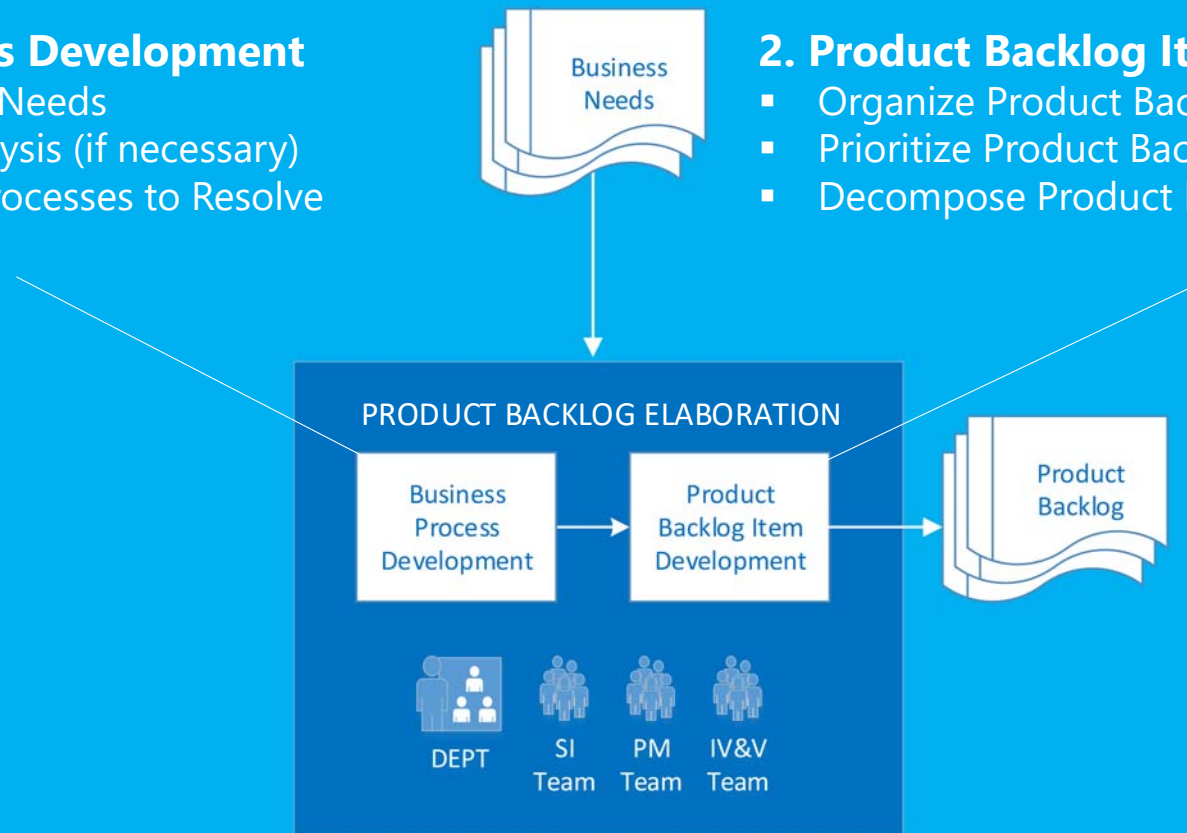
# STATE DEVELOPMENT FRAMEWORK (SDF)

## 1. Business Process Development

- Evaluate Business Needs
- Conduct Gap Analysis (if necessary)
- Define Business Processes to Resolve Gaps

## 2. Product Backlog Item Development

- Organize Product Backlog Items
- Prioritize Product Backlog Items
- Decompose Product Backlog Items



# USER STORY DEVELOPMENT

- Identify Epics (large user stories)
  - Example: Apply for a license
- Identify Themes (collection of user stories)
  - Example: Monthly Report Formatting
- Develop user stories using INVEST (independent, negotiable, valuable, estimable, small, testable)
  - Example: As an applicant, I would like to set my password, so I can log into the system.

# STATE DEVELOPMENT FRAMEWORK (SDF)

1. Sprint Planning

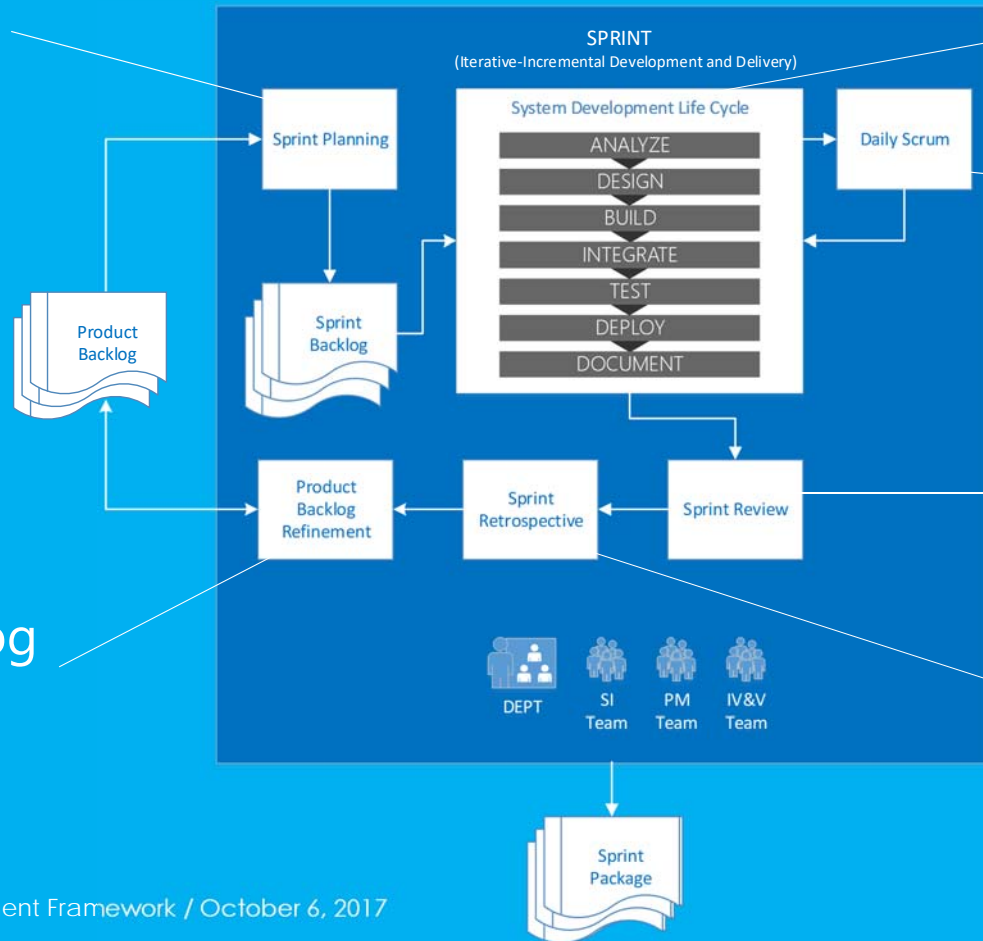
2. Sprint Cycle

3. Daily Scrum

4. Sprint Review

5. Sprint Retrospective

6. Product Backlog Refinement





# SPRINT PACKAGE

- Product Backlog
- Sprint Backlog
- Product Increment
- Acceptance Criteria
- Test Scripts
- Test Results Report
- Sprint Review Meeting Minutes
- Sprint Retrospective Meeting Minutes
- Sprint Status Report
- Project Management Documentation

# STAKEHOLDERS AND CONSULTANTS

- Department
- SI Team
- PM Team
- IV&V Team
- External Entities (i.e., Department of Justice)



DEPT



SI  
Team



PM  
Team



IV&V  
Team

# SCRUM TEAM

- Product Owner(s)
- Scrum Master
- Scrum Development Team
  - Developer(s)
  - UX Designer(s)
  - Business Analyst(s)
  - Quality Assurance Analyst(s)
  - Subject Matter Expert(s)
  - Tester(s)



# PRODUCT OWNER(S)

- Responsible for product vision
- **Constantly re-prioritizes the Product Backlog**
- Accepts or rejects each product increment
- **Final decision-maker of requirements questions**
- Decides whether to continue development
- Considers stakeholder interests
- May contribute as a team member
- **Has a leadership role and from the Department**



# SCRUM MASTER

- **Facilitates the Scrum process**
- Helps resolve impediments
- **Creates an environment promoting team self-organization**
- Captures empirical data to adjust forecasts
- Shields the team from external interference
- Enforces timeboxes
- Keeps Scrum artifacts visible
- Has no management authority over the team
- **Has a leadership role and on the SI Team**



# SCRUM DEVELOPMENT TEAM

- **Cross-functional**
- Negotiates commitments with the Product Owner, one Sprint at a time
- Has autonomy regarding how to reach commitments
- **Intensely collaborative**
- Most successful when located in one team room, particularly for the first few Sprints
- **Self-organizing and Self-managing**
- Contains an average of 5 to 9 members with the majority from the SI Team



# Concerns with SDF

THE CHALLENGE OF TRANSITIONING FROM A TRADITIONAL  
CULTURE AND AN AGILE CULTURE

# WHAT TYPE OF CONTRACT FITS WITH SDF?

- Time and Materials Contract
- Work Order Authorizations (WOAs)
  - Deliverables
  - Work Products
  - Activities
- Sprints and Releases
  - Minimum Viable Product
  - UAT and Go-Live



# WHAT PLANS WILL FIT WITH SDF?

- California Project Management Framework (CA-PMF)
- Software Development Plan
- Contract Management Plan
- Governance Management Plan
- Deliverables Management Plan
- Change Control Management Plan
- Requirements Management Plan
- Risk and Issue Management Plan
- Test and Evaluation Management Plan
- Maintenance and Operations Plan

# HOW DO WE PASS PAL WITH SDF?

- SIMM 19 – Project Approval Lifecycle (PAL)
- Stage 1 – Business Analysis
- Stage 2 – Alternatives Analysis
- Stage 3 – Solution Development
- Stage 4 – Project Readiness
- Financial Analysis Worksheets (FAWs)

# HOW DO WE CONDUCT TESTING?

- Test Driven Development
- Regression or Validation Testing
- User Acceptance Testing
- Performance Testing
- Security Testing
- Section 508 Testing



Questions?

# REFERENCES

- Agile Alliance - <https://www.agilealliance.org/>
- Agile Manifesto - <http://agilemanifesto.org/>
- CDT SIMM 19 - <https://cdt.ca.gov/policy/simm-19/>
- CDT CA-PMF - <http://capmf.cio.ca.gov/>
- LeSS - <https://less.works/>
- SAFe - <http://www.scaledagileframework.com/>
- Scrum Alliance - <https://www.scrumalliance.org/>