



VCU

College of Engineering

CS 25-327 XState Builder

Project Proposal

Prepared for

Mahesh Nair & Jaquelyn Dellinger

Capital One

By

Sohum Dharamsi

Neil Randori

Bryan Wheeler

Thien Dang

Under the supervision of

Irfan Ahmed

Date

October 11th, 2024

Executive Summary

This project aims to develop a web application that facilitates the creation of XState files to define customers' event-driven workflows. The platform will enable call center agents to fill out an intake form via Slack, generating XState definition files based on their responses.

The primary objectives are as follows:

1. Web Application Development
 - a. Create a user-friendly interface for displaying available XState definitions
 - b. Enable users to trigger events within each XState definition.
2. Database Integration
 - a. Establish database tables to manage relevant data, associating tasks with specific workflow instances.
3. Slack Integration
 - a. Design an intake form within Slack to collect user input.
 - b. Ensure the form poses relevant, user-friendly questions to simplify the creation of XState definition files.
 - c. Automate the opening of pull requests in a GitHub repository, reflecting the newly generated XState definitions.

This project will streamline the process of defining customer workflows allowing agents to more effectively manage a higher volume of cases.

Table of Contents

Section A. Problem Statement	5
Section B. Engineering Design Requirements	7
B.1 Project Goals (i.e. Client Needs)	7
B.2 Design Objectives	7
B.3 Design Specifications and Constraints	8
B.4 Codes and Standards	9
Section C. Scope of Work	11
C.1 Deliverables	11
C.2 Milestones	12
C.3 Resources	12
Section D. Concept Generation	13
Section E. Concept Evaluation and Selection	14
Section F. Design Methodology	16
F.1 Computational Methods (e.g. FEA or CFD Modeling, example sub-section)	16
F.2 Experimental Methods (example subsection)	16
F.5 Validation Procedure	16
Section G. Results and Design Details	18
G.1 Modeling Results (example subsection)	18
G.2 Experimental Results (example subsection)	18
G.3 Prototyping and Testing Results (example subsection)	18
G.4. Final Design Details/Specifications (example subsection)	18
Section H. Societal Impacts of Design	20
H.1 Public Health, Safety, and Welfare	20
H.2 Societal Impacts	20
H.3 Political/Regulatory Impacts	20
H.4. Economic Impacts	20
H.5 Environmental Impacts	21
H.6 Global Impacts	21

H.7. Ethical Considerations	21
Section I. Cost Analysis	22
Section J. Conclusions and Recommendations	23
Appendix 1: Project Timeline	24
Appendix 2: Team Contract (i.e. Team Organization)	25
Appendix 3: [Insert Appendix Title]	26
References	27

Section A. Problem Statement

Industry Context

The project falls within the field of customer service management and workflow automation, specifically focusing on call centers. Call centers are critical in delivering customer support across various industries, including telecommunications, finance, healthcare, and e-commerce. The effectiveness of a call center is largely determined by its ability to handle customer inquiries promptly and accurately. As businesses increasingly rely on technology to enhance service delivery, the demand for efficient, event-driven workflows has grown.

Identifying the Problem

Call center agents often encounter challenges when defining and managing workflows for handling customer inquiries and events. Traditional methods for creating workflow definitions, often involving manual processes or complex coding, can be time-consuming and error-prone.

This inefficiency can lead to:

1. Long Response Times: Slow workflow design leads to delays in addressing customer needs, impacting overall satisfaction.
2. Increased Training Costs: New agents require extensive training to understand existing workflows, which can vary significantly across different systems.
3. Lack of Flexibility: Static workflow definitions can struggle to adapt to evolving customer needs, resulting in lost opportunities for service improvement.

Prevalence and Costs of the Problem

This issue is widespread in the customer service industry. Many call centers still rely on outdated workflow management systems that do not integrate well with modern communication platforms like Slack.

The costs associated with these inefficiencies can be substantial, including:

1. Economic Costs: Reduced customer retention rates, resulting in lost revenue. Studies show that a 5% increase in customer retention can increase profits by 25% to 95%.
2. Societal Costs: Poor customer service experiences can lead to customer frustration and dissatisfaction, affecting brand reputation and customer loyalty.
3. Health and Safety Costs: In high-pressure environments, such as call centers, inefficiencies can contribute to employee stress and burnout, leading to increased turnover rates.

Project Client and Stakeholders

The primary client for this project is the platform stakeholder (Capital One) seeking to improve the efficiency of their call center operations. Key stakeholders include call center agents, workflow managers, and IT departments responsible for implementing and maintaining the technology stack. Additionally, customers who interact with the call center indirectly benefit from improved workflows and quicker response times.

Historical Perspective and Previous Solutions

Historically, workflow automation in call centers has involved complex coding or the use of dedicated workflow management software. Solutions like BPM (Business Process Management) tools have emerged, but they often lack integration capabilities with communication platforms and can be cumbersome to use for non-technical staff.

Previous attempts to streamline workflow design included:

1. Low-Code/No-Code Platforms: Tools like Zapier and Airtable allow users to automate tasks without deep coding knowledge. However, they often lack the specific features needed for event-driven workflows.
 - a. Pros: User-friendly and accessible.
 - b. Cons: Limited flexibility for complex workflows.
2. Traditional BPM Solutions: Software like IBM BPM or Appian provides robust workflow management capabilities.
 - a. Pros: Highly customizable.
 - b. Cons: Requires significant training and is often too complex for everyday users.
3. Custom Solutions: Some organizations develop bespoke systems to manage workflows.
 - a. Pros: Tailored to specific needs.
 - b. Cons: High development and maintenance costs.

Current Project and Innovations

This project aims to improve upon existing solutions by creating an easy-to-use web application that integrates seamlessly with Slack and automates the generation of XState definitions. This approach addresses several unmet needs:

1. Ease of Use: By utilizing a user-friendly intake form, call center agents can generate workflow definitions without needing technical expertise.
2. Integration with Modern Tools: Leveraging Slack as a communication platform allows for real-time interaction and data collection.

In conclusion, the proposed project aims to fill a significant gap in the market by providing a solution that combines ease of use with robust workflow automation capabilities, ultimately advancing customer service technology and improving operational efficiency in call centers. By addressing the identified challenges, this initiative seeks to enhance both agent performance and customer satisfaction, contributing positively to the broader industry landscape.

Section B. Engineering Design Requirements

B.1 Project Goals (i.e. Client Needs)

The primary goal of this project is to develop a web application that enables stakeholders to efficiently create, manage, and review event driven workflows using XState. This application will capture user requirements, automate workflow processes, and ensure integration with various tools like Slack and Github.

- Streamlined workflow creation based on user needs
- Integration of state machines for reliable task management
- User-friendly interface to both technical and non-technical users

B.2 Design Objectives

List the key objectives of the design that you will produce. Objectives describe *what the design will do*, not how it should do it. Objectives should be SMART – Specific, Measurable, Achievable, Realistic, and Time-bound. Each objective will ultimately be linked to a design specification/constraint during the design process. Again, lists are nice if applicable.

- Workflow Management: Allows users to create, review, and manage event-driven workflow efficiently
- Task Tracking: Tasks are tracked and updated in real-time within each workflow instances
- Integration: Ensure seamless integration of Slack and Github
- State transition: Incorporate XState to manage precise and predictable state transitions in a workflow
- Data Management: Postgres is used to retrieve and store instances and task data

B.3 Design Specifications and Constraints

A list of design specifications and constraints include all limitations, restrictions, and requirements of the design. They are firm limits that must be met for a design to be acceptable and are ultimately used to measure the success of a design. Each specification or constraint should map to one or more design objective(s) and explicitly state *how the design* will meet the objectives. Specifications and constraints should be specific and are often numerical. They must be measurable or testable to prove that the design has met all of the design objectives. Numerical metrics may include qualifying statements such as “at least,” “at most,” “between,” “exactly” or include a set of discrete values. Avoid subjective, untestable constraints (e.g. “environmentally friendly”, “user friendly”, “nice looking”, etc.).

Each design objective is mapped to a specification:

- Workflow Management: at least three actions available to user to create, review, and manage cases
- Task Tracking: Software allows for one list of tasks in the user interface that updates as tasks are completed
- Integration: Slack and Github are connected to the software and each case will have exactly one representation in both Slack and Github
- State transition: Each case will have exactly one Xstate file
- Data Management: Postgres will retrieve and store each case

Realistic constraints can come take on a variety of forms including accessibility, aesthetics, codes, constructability, cost, ergonomics, extensibility, functionality, interoperability, legal considerations, maintainability, manufacturability, marketability, policy, regulations, schedule, standards, sustainability, or usability. Examples of physical constraints might include numerical limits or ranges on overall size envelope, weight, pressures, stresses, flow rates, voltages, current, power consumption, hardware limitations, data constraints, interoperability, etc. Other constraints might include production unit cost, expected part/device life, or maintenance requirements.

Constraints are as follows:

- Software must integrate into Slack and Github
- Software must smoothly integrate into Capital One's software system
- Format of management must follow Capital One case management standards
- Software should follow the correct order of case management provided by Capital One's standards (Open > review > close, etc.)

Section C. Scope of Work

The project scope defines the boundaries of the project encompassing the key objectives, timeline, milestones and deliverables. It clearly defines the responsibility of the team and the process by which the proposed work will be verified and approved. A clear scope helps to facilitate understanding of the project, reduce ambiguities and risk, and manage expectations. In addition to stating the responsibilities of the team, it should also explicitly state those tasks which fall *outside* of the team's responsibilities. *Explicit bounds* on the project timeline, available funds, and promised deliverables should be clearly stated. These boundaries help to avoid *scope creep*, or changes to the scope of the project without any control. This section also defines the project approach, the development methodology used in developing the solution, such as waterfall or agile (shall be chosen in concert with the faculty advisor and/or project sponsor). Good communication with the project sponsor and faculty advisor is the most effective way to stay within scope and make sure all objectives and deliverables are met on time and on budget.

C.1 Deliverables

The project deliverables are those things that the project team is responsible for providing to the project sponsor. They are the things that are to be produced or provided as a result of the engineering design process. Some deliverables might include a specific number of alternative designs, required analyses to prove the design meets specifications, detailed machine drawings, functional diagrams or schematics, required computer code, flow charts, user manuals, desktop models, and functioning prototypes. A design “proof of concept” is not specific and should be more clearly defined. Academic deliverables include the team contract, project proposal, preliminary design report, fall poster and presentation, final design report, and Capstone EXPO poster and presentation. Provide a bulleted list of all agreed upon project deliverables.

Deliverables:

- Landing Page
 - Workflow page
 - Database developed
 - Slack Integration
 - Github Integration
 - Final interface
-
- All work can be done remotely, with consistent communication from team members

C.2 Milestones

- Configuration of initial vue.js application and docker container
- Creation of landing page
- Creation of workflow page
- Design of database schema
- Configuration of PostgreSQL database
- Slack integration / design of intake form
- Github integration
- Testing

Milestones are major project phases or tasks that need to be completed in order to ensure the project deliverables. They may include, among other things, completion of calculations, the development of a computational model, completion of an analysis, set-up of an experiment, completion of data acquisition, purchasing of hardware, assembly of a prototype, completion of testing procedures, development of required code, completion of wiring, post processing, etc.

A good rule of thumb is to break the project down into tasks of no larger than 2-3 weeks in length. These can be individual or group tasks. Breaking down the project into tasks/milestones gives the team and the advisor/sponsor a realistic understanding of what can be done in the allotted time. In an agile development approach, later tasks are expected to be adjusted (or changed) as the team works with the earlier developed tasks.

The amount of time it will take to accomplish each milestone and the approximate date that each milestone will be completed should be considered. Do not underestimate the time that it takes to write and prepare major reports and presentation materials. All deliverables and milestones should be included in the project timeline found in Appendix 1. Provide a summary table of all project milestones including required times and completion dates here.

Note: While the project scope, deliverable, and milestones are not intended to change throughout the project, this section should be revisited between major reports to ensure that it still accurately reflects the expectations and requirements of the project team, client, and faculty advisor. Any changes to the project scope, deliverable, and milestones should be thoroughly discussed and mutually agreed upon by all parties. Any changes to this section should be documented and justified in detail.

C.3 Resources

Resources needed for project completion should be listed at the proposal stage. These resources can either be purchased within the Project Budget, or provided by the project sponsor. Some examples are: hardware such as HPCs or servers, software such as IDEs, data analysis platforms or version control systems. Access to cloud computing services may also be necessary

to scale certain procedures. Additionally, databases containing operational data for testing, as well as libraries or APIs relevant to predictive analytics and machine learning may be required.

- Some resources required by the team:
 - Data or mock data from Capital One to test software

Section D. Concept Generation

A number of methods can be used to help generate design concepts from simple reflection and brainstorming, to working the problem backwards, using reverse thinking techniques, and looking to nature for inspiration (i.e. biomimicry). Existing solutions, or components of existing solutions, can be substituted, combined, adapted, modified, put to other uses, eliminated, or rearranged to meet new design objectives and specifications. A minimum of 3 overall design concepts is required for this section although more are welcome. Provide a brief description of how each design concept addresses the design problem. Discuss the potential pros and cons, including and potential risks of failure, of each of these concepts.

It is likely that each design concept may consist of several components. In this case, one or more of these components may offer a sub-problem that can be further explored, modified, or otherwise improved upon. These sub-problems may lead to the addition of several additional design concepts and may require the inclusion of a design concept chart or matrix to organize all ideas and potential solutions.

Provide any initial design sketches, drawings, 3D renderings, or conceptual models such as dataflow diagrams, process flows, etc. developed during the concept ideation phase. All hand drawings should be drawn to scale using basic engineering drafting tools (i.e. ruler, protractor, and compass). Geometric stencils can also be used to help produce quality hand drawings. Drawings should be presented in a professional manner, preferably done on engineering graph paper and using a high-quality scan. All sketches should be labeled to identify major components and different drawing views or projections if applicable. Basic dimensions should be provided to give a general sense of scale. Label each sketch or drawing with the name of the team member responsible for the sketch, the date it was drawn, and the drawing scale.

Section E. Concept Evaluation and Selection

Using a systematic decision-making process, evaluate each of the design concepts and choose the one that is most likely to succeed in meeting the design objectives and constraints. A Decision Matrix, or Pugh Matrix, helps to analyze alternatives, eliminate biases, and make rational decisions through thought and structure. First, work to develop a set of selection criteria for which to evaluate the previously generated design concepts. Selection criteria often include concepts of performance, cost, safety, reliability, risk, etc. Note that the selection criteria developed here will likely be more general than the project design objectives. As with the design objectives, conversations with the client help define appropriate selection criteria.

In many cases, the client may value the selection criteria differently, preferring that more emphasis be placed on some than others. In this case, weighting factors may be used to place more or less importance on the various criteria in the decision making process. Again, conversations with the client can be used to define criteria weighting factors. Often times, these conversations must be analyzed and interpreted by the team to determine which criteria are more important to the client and by how much. Feel free to discuss the assigned weighting factors with the client to see if they seem accurate.

Next, define an associated metric to represent each criteria. Metrics should be specific and quantifiable, providing numerical values that quantify the often vague concepts of the selection criteria. Metrics can be obtained, generated, or estimated through a number of methods including simple background research, preliminary design calculations, or basic analyses. Note that these metrics do not need to specifically align with the design specifications although there may be some commonality between the two. Provide a brief discussion of the rationale for selecting each of the assigned metrics.

Using the defined metrics, evaluate each design concept against all selection criteria by filling out a Decision Matrix. Design concepts can be compared by using simple rank scoring, raw scoring, or weighted scoring techniques and design concept with which to move forward can be selected. This type of process provides a meaningful, unbiased means for choosing a preliminary design concept prior to moving forward with more comprehensive, detailed analyses as provided in the design methodology section below. The results of this process should be discussed with the project client prior to moving forward with the selected design. Table 1 provides an example of a simple decision matrix.

Table 1. Example of a Decision Matrix.

	Design Concept A	Design Concept B	Design Concept C	Design Concept D
Criteria 1				
Criteria 2				
Criteria 3				
Criteria 4				
Criteria 5				
Total Score				

Note: Weights can be assigned to each criterion if desired.

Section F. Design Methodology

Provide a detailed explanation of the methods that will be used to help evaluate, improve, and evolve the design through the iterative engineering design process. Consider that ultimately, the final design must be verified and validated to ensure that it meets all of the previously developed and listed design objectives and specifications. Verification ensures that the design meets all specification, while validation confirms that the design functions as intended such to meet the client's needs. While it is common for initial design concepts to first be evaluated using simplified design criteria and metrics, the chosen design should be advanced, and later verified, using engineering calculations, computational models, experimental data, and/or testing procedures.

Use this section to describe any underlying physical principles and mathematical equations that govern the design. Provide details of any computer-aided modeling techniques used to evaluate the design including the software used, prescribed boundary conditions, and assumptions. Include a detailed description of any experimental testing methods including required testing equipment, test set-up layout, data acquisition and instrumentation, and testing procedures. If one or more prototypes is to be produced and tested, provide a detailed description of how each will be evaluated.

Note: The contents of this section are expected to vary from project to project. Subsections may be appropriate for providing details of analytical, computational, experimental, and/or testing methods. Some potential subsections that may be included in this section are provided. While critical design equations may be provided here, lengthy mathematical derivations may be included in an appendix. Validation procedures are critical and all projects should address such topic.

F.1 Computational Methods (e.g. FEA or CFD Modeling, example sub-section)

F.2 Experimental Methods (example subsection)

F.3 Architecture/High-level Design (example subsection)

F.5 Validation Procedure

Describe how the design team will validate that the final design meets the client's needs. This section should include a plan to meet with the client towards the end of the project to discuss final design details and demonstrate a prototype, experimental test, and/or simulation results. Provide a relative time frame for this validation to occur (e.g. "mid-March" or "early-April"). Include a brief discussion on how client feedback will be captured, such as a

formal survey, interview, or observation notes of the client using the prototype. It may also include plans to solicit feedback from other stakeholders and/or potential users.

Section G. Results and Design Details

Use this section to highlight the major results of the design methodology described above including important analytical, computational, experimental, modeling, assembly, and testing results. This section should be one of the most substantial sections of the report showcasing all of the hard work and effort that went into the completion of the final design and delivery of the project deliverables. Show how the identified problem was solved.

Highlight the prominent features of the final design through analysis results, modeling, drawings, renderings, circuit schematics, instrumentation diagrams, flow and piping diagrams, etc. to show that the design functions as intended and meets all design objectives and constraints. Overview designs such as dataflow diagrams, process flow, swim lane diagrams, as well as presentation-layer designs (e.g. storyboards for front-ends) should be included here. Detailed designs such as database designs, software designs, procedure flowcharts, or pseudocode should be included here. Support computational and experimental results with key plots and figures. All supporting figures should be clearly labeled and annotated to highlight the most important points of the figure (i.e. explicitly point out what the reader should focus on or understand about the image).

Note that while all results should be used to help inform design decisions, not all results may be necessary to include in the main body of the report. Extraneous supporting results (e.g. graphs, data, design renderings, drawings, etc.) that are not necessary for presenting the fundamental findings can be placed in one or more appendices. Detailed documentation of each program module can be provided as appendix.

G.1 Modeling Results (example subsection)

G.2 Experimental Results (example subsection)

G.3 Prototyping and Testing Results (example subsection)

G.4. Final Design Details/Specifications (example subsection)

Note that while the design constraints and specifications may have provided minimum or maximum values, or ranges or values, that the design needed to meet, the final design specifications should be listed here showing that the required design values were met. A list of final design details can also be included demonstrate fulfillment of the design objectives.

Note: Preliminary results should be included in the Preliminary Design Report to show the progress made of the selected design concept to-date. This section should be updated for the Final Design Report to include documentation of all of the work that was completed on the project throughout the entirety of the academic year.

Section H. Societal Impacts of Design

In addition to technical design considerations, contemporary engineers must consider the broader impacts that their design choices have on the world around them. These impacts include the consideration of public health, safety, and welfare as well as the potential societal, political/regulatory, economic, environmental, global, and ethical impacts of the design. As appropriate for the project design, discuss how each of these considerations influenced design choices in separate subsections. How will the design change the way people interact with each other? What are the political implications of the design? Does the technology have the potential to impact or shift markets? Does the design have any positive or negative effects on the environment? Don't forget to consider unintended consequences such as process or manufacturing byproducts. What impacts might the design have on global markets and trade? Are there any ethical questions related to the design?

While it is hard to forecast the various impacts of a technology, it is important to consider these potential impacts throughout the engineering design process. When considered during the early stages of the design phase, consideration of these impacts can help determine design objectives, constraints, and specifications and help drive design choices that may mitigate any potential negative impacts or unintended consequences.

Note: A minimum of 4 of these design considerations, including the consideration of public health, safety, and welfare, are required for the Preliminary Design Report while a section for all considerations must be included in the final design report.

H.1 Public Health, Safety, and Welfare

Provide a list of all design safety features and provide a brief description of each. Discuss the potential effects the design may have on public health, safety, and welfare. References to the codes and standards previously provided and the organizations that produced them may be summarized or referenced here.

H.2 Societal Impacts

H.3 Political/Regulatory Impacts

H.4. Economic Impacts

H.5 Environmental Impacts

H.6 Global Impacts

H.7. Ethical Considerations

Section I. Cost Analysis

Provide a simple cost analysis of the project that includes a list of all expenditures related to the project. If an experimental test set-up or prototype was developed, provide a Bill of Materials that includes part numbers, vendor names, unit costs, quantity, total costs, delivery times, dates received, etc. Do not forget to include all manufacturing costs incurred throughout the completion of the project. If the design is expected to become a commercial product, provide a production cost estimate including fixed capital, raw materials, manufacturing (including tooling and/or casting), and labor costs to produce and package the device. Note that this type of detailed cost analysis may be listed as a project deliverable.

Note: The Preliminary Design Report should include all costs incurred to date. It is expected that this section will be expanded and updated between the preliminary and final design reports.

Section J. Conclusions and Recommendations

Use this section to summarize the story of how the design team arrived at the final design. Focus on the evolution of the design through the use of the engineering design process including lessons learned, obstacles overcome, and triumphs of the final design. Revisit the primary project goals and objectives. Provide a brief summary of the final design details and features paramount to the function of the design in meeting these goals and objectives.

A discussion may be included to discuss how the design could be further advanced or improved in the future. If applicable, summarize any questions or curiosities that the final results/design of this effort bring to mind or leave unanswered. If this project might continue on as a future (continuation) senior design project, detail the major milestones that have been completed to date and include any suggested testing plans, relevant machine drawings, electrical schematics, developed computer code, etc. All relevant information should be included in this section such that future researchers could pick up the project and advance the work in as seamless a manner as possible. Documents such as drawings, schematics, and codes could be referenced here and included in one or more appendix. If digital files are critical for future work, they should be saved on a thumb drive, external hard drive, cloud, etc. and left in the hands of the project advisor and/or client.

Appendix 1: Project Timeline

Provide a Gantt chart of similarly composed visual timeline showing the start and end dates of all completed tasks and how they are grouped together, overlapped, and linked together. Include all senior design requirements including design reports and Expo materials (i.e. Abstract, Poster, and Presentation). All major milestones should be included in the timeline.

Appendix 2: Team Contract (i.e. Team Organization)

Copy and paste the content from the completed Team Contract here starting with Step 1 of the Team Contract and including all content following the 'Contents' list.

Appendix 3: [Insert Appendix Title]

Note that additional appendices may be added as needed. Appendices are used for supplementary material considered or used in the design process but not necessary for understanding the fundamental design or results. Lengthy mathematical derivations, ancillary results (e.g. data sets, plots), and detailed mechanical drawings are examples of items that might be placed in an appendix. Multiple appendices may be used to delineate topics and can be labeled using letters or numbers. Each appendix should start on a new page. Reference each appendix and the information it contains in the main text of the report where appropriate.

Note: Delete this page if no additional appendices are included.

References

Provide a numbered list of all references in order of appearance using APA citation format. The reference page should begin on a new page as shown here.

- [1] VCU Writing Center. (2021, September 8). *APA Citation: A guide to formatting in APA style*. Retrieved September 2, 2024. <https://writing.vcu.edu/student-resources/apa-citations/>
- [2] Teach Engineering. *Engineering Design Process*. TeachEngineering.org. Retrieved September 2, 2024. <https://www.teachengineering.org/populartopics/designprocess>