

Zenatix

Testing SOW - Proposal

7/29/2019
FutureSoft India Private Limited
Version 1.1

TABLE OF CONTENTS

Background	2
System Understanding	2
Architecture Diagram	2
Key Notes	3
Offerings	3
Development of Simulation modules	3
Rich Visualization & Data Preparation	4
System/Functional/Integration/Regression/Stress Testing	4
Example Test Scenarios	4
Security Testing & Documentation	5
Rules Documentation & Scripts for testing	5
Standardization of rollout processes	5
Detection of Compatibility Issues – H/W, S/W, Builds, Protocols	5
Help Fix Support Issues - Python 2 Support Discontinuation	6
DevOps Process Improvement	6
Engagement Model Options - Suggested	6
Resource Augmentation	6
Solution Center Set up @FutureSoft	7
Commercials	8
Resource Augmentation	8
Solution Center Set up @FutureSoft	8
Final Engagement Model (Discussion Dated – 23 July)	9
Payment Terms	9

Zenatix – Testing SOW

BACKGROUND

Zenatix engaged FutureSoft to analyse their IOT platform for suggesting gaps in the testing processes. FutureSoft team had two sessions with Zenatix team to understand the processes & system components.

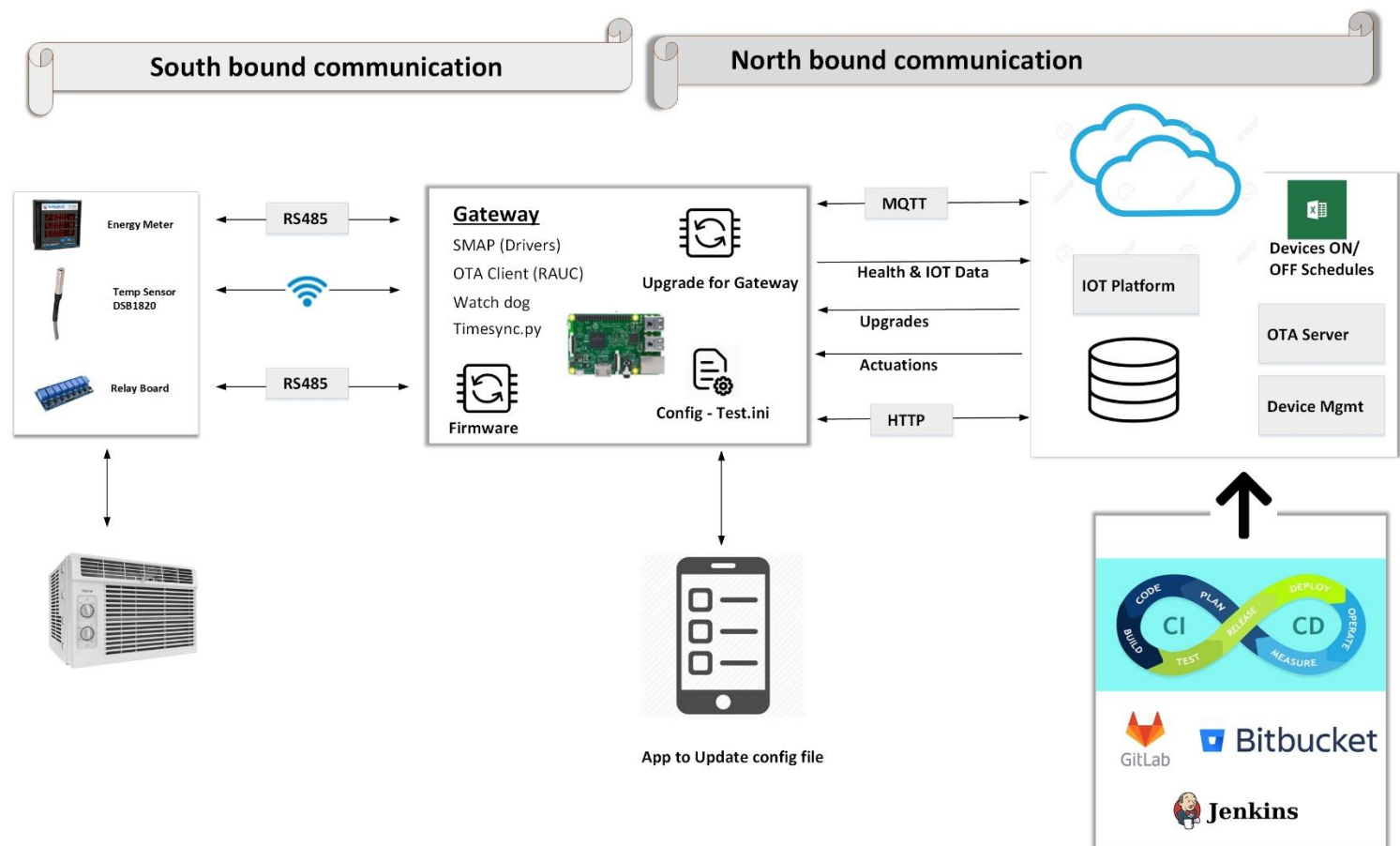
Objective of these sessions was to understand current testing processes being followed by Zenatix and to identify existing pain points so that appropriate recommendations for engagement can be suggested.

Sessions were held on 3rd July & 10th July.

SYSTEM UNDERSTANDING

ARCHITECTURE DIAGRAM

Zenatix Platform Architecture (Understanding)



Zenatix – Testing SOW

Platform is being used to sense two kinds of data.

- Energy
- Temperature

In addition, it is being used for actuation of devices such as ACs, Ovens etc.

Control layer of platform is referred to as **Gateway**, aka, **Edge**. It acts as bridge between sensors and cloud stack.

Communication between **sensors & edge/gateway** is referred as **South bound communication**.

Communication between **edge/gateway & cloud** is referred as **North bound communication**.

KEY NOTES

- Actuations are pushed from cloud to gateway and then to relay/actual devices to trigger ON/OFF/Reboot actions.
- IOT/Business relevant data is pushed from sensors to gateway and then to IOT platform on cloud stack.
- **SMAP – Simple measurement actuation protocol** – runs in each RPi and sends data to IOT Platform on cloud. It is built on top of **twisted** framework. It implements drivers for various devices. These drivers read configuration file to know about devices, read data from these devices and transmits the data to cloud.
- SMAP configuration file is bundled initially in the deployable package. Subsequent updates to it can be made through a supplementary phone app.
- Replication of time interval defined in configuration file is not instantaneously replicated on devices.
- OTA client checks for upgrades by sending device id to Hawkbit server.
- Health data is pushed from gateway to device management module on cloud stack.
- ON/OFF timings per site per controller are uploaded into the system (AWS Instance) through an excel sheet.
- Time series data is stored in Readingdb.
- In OTA upgrade, software packages are pushed from cloud (OTA Server) to Raspberry Pi (OTA Client). Software packages may include Python scripts, Debian, Firmware packages etc.
- End Points can buffer data for 12 Hours.
- The meter (Elmeasure LG+ 5310) is being used as an endpoint in the IOT solution. Meter is connected with Raspberry Pi through USB to RS485 over Modbus.
- Relay board is being used to receive commands from the Cloud in real time.
- ESP8266 is being used in Station Mode (To connect to Zenatix Network) as well as Access Point Mode (For local configuration).

OFFERINGS

DEVELOPMENT OF SIMULATION MODULES

Current Solution is heavily dependent on availability of data points from various hardware interfaces. It is extremely important that sufficient data points are readily available for integration & stress testing of all the interfaces.

There is serious need of data simulation modules. FutureSoft team can develop these modules for various hardware interfaces. While developing these simulation modules, some of the parameters stated below would be taken care of. Additional parameters can be added based on mutual agreement.

- Modes of communication (Wi-Fi, Bluetooth)
- Protocols (HTTP, COAP, MQTT)
- Device Types (Sensors, Energy Meters, RPi)

Zenatix – Testing SOW

- Write operations limit of SD cards,
- Capacity of SD cards,
- CPU usage
- Bandwidth Consumption

RICH VISUALIZATION & DATA PREPARATION

FutureSoft can suggest multiple ways to represent data in charts. Since dashboard is critical part of the platform. Our team can suggest visually appealing charts & dashboard for effective representation of facts.

For better visualizations, adequate data should be available. More importantly, for new implementations there is always dearth of data. Data preparation (along with rich visualizations) is key factor in success of new implementations. This gaps can be filled by FutureSoft.

SYSTEM/FUNCTIONAL/INTEGRATION/REGRESSION/STRESS TESTING

FutureSoft can contribute in various kinds of testing requirements. Test defects can be logged in JIRA & tracked for execution. Status reporting of tests – such as open, in-progress, completed etc., can be shared on agreed interval.

EXAMPLE TEST SCENARIOS

- Rules Testing - Special Days Testing
- Upgrades Testing –
 - Packages are upgraded or not.
 - SMAP rollouts are working fine or not.
- Heartbeat Testing - Services are responding fine.
- Scheduled actions
- Boundary Tests (Sensor Values)
- Control Tests (ON/OFF)
- Exceptions - Voltage Drop/Fluctuations
- Priority clashes (Time/Rule Exception)
- Test Average Mean life for failure detection (SD Card Write operations)
- Latency over various protocols
- Bandwidth (Before/After upgrade, Impact on it with increase/decrease in sensors)
- Versioning/Compatibility Issues
- CPU Usage
- Number of reboots
- RAM Usage
- Wi-Fi signal strength
- Wi-Fi connect/disconnect counter
- Write Operations limit testing
- Stress Testing - FutureSoft would help in building an environment which would virtualize the end point for Raspberry Pi. With the help of virtualization, we would be able to perform stress testing from the end point side.

Zenatix – Testing SOW

SECURITY TESTING & DOCUMENTATION

Security is always an important factor in success of any platform. There is potential risk of security breach at each layer of an IOT platform. FutureSoft team can analyse all the layers & tiers of the platform to identify potential risks and suggest recommendations.

Areas where we can help are,

- Test application modules/components for security. (Periodic activity)
- Identity/Access management for each tier of the platform.
- Data Injection/Validation issues testing
- Help formalize recovery plans in case of cyber attack

RULES DOCUMENTATION & SCRIPTS FOR TESTING

During discussions we gathered that actuations & data collection in the platform are dependent on pre-configured rules. Some of the rules are time based, example ON/OFF timings, periodicity of data collection etc.

Other rules are based on device status/conditions, example, If voltage remains <40 or >270 for more than 2 minutes then switch off all ACs or Ovens ON should trigger Exhaust fans ON. In addition, there are rules for special days etc.

FutureSoft team can understand, explore and document all the test cases and generate scripts/test data which can trigger failure of key events due to clash of priorities. It would help Zenatix team to proactively handle such issues and avoid hefty losses caused due to wrong/missed triggers. This information/document can act as single source of information for various teams such as Dev, test & Ops team.

Later, if there is a need, we can recommend hosting these rules in appropriate rule engine platform. There are plenty of rule engine platforms available in market. These engines comes with intuitive user interfaces for rules authoring and provides mechanism to integrate with existing platforms by exposing rules as web services.

STANDARDIZATION OF ROLLOUT PROCESSES

Rollout of upgrades is critical factor in success of the solution. This process can be strengthened by reviewing current rollout process and documenting the same.

It is expected to discover various gaps related with versioning, compatibility, sizing, security, support and privacy issues.

Further, it has to be a living document/process so that it remains current.

Once this structure is in place, it can act as reference point for future rollouts. Further, this can be consumed by DevOps team to make improvement in existing CI/CD processes.

DETECTION OF COMPATIBILITY ISSUES – H/W, S/W, BUILDS, PROTOCOLS

Current solution is built on top of multiple open source frameworks (SMAP, Hawkbit, Twisted). Further, the solution provides interfacing between various hardware components over different protocols and communication media.

There is strong probability of compatibility issues due to either of below reasons,

- Different versions of software & protocol requirement for different hardware,
- Rollout of a build on unsupported hardware

Zenatix – Testing SOW

- Build deployed with incorrect version of packages

FutureSoft team can help in streamlining documentation & testing of compatibility issues.

HELP FIX SUPPORT ISSUES - PYTHON 2 SUPPORT DISCONTINUATION

Edge stack deployed on Raspberry Pi is divided into two parts:

SMAP based on Twisted

- Twisted is an event driven Python 2 framework using which SMAP (application) is written.
- Deployed on Raspberry Pi
- The event drives a call back either on sensors or on cloud or both

Hawkbit Server and RAUC

- Hawkbit is java based Eclipse Project for Embedded device software rollouts
- RAUC is Python 3 based Hawkbit Client (on Raspberry Pi) which connects to Hawkbit Server) deployed on Raspberry Pi

The support for Python 2 is ending in January 2020, hence Twisted. Therefore, the custom packages (drivers - python scripts) for the Zenatix would have to be manually migrated to Python 3.

FutureSoft team can analyse probability of issues happening due to Python 2 support discontinuation and can help Zenatix take decision in proactive manner rather than reactive manner.

DEVOPS PROCESS IMPROVEMENT

FutureSoft can understand existing DevOps practices and help streamline the same. Existing CI/CD pipelines can be reviewed and improvement recommendations can be implemented by FutureSoft/shared with Zenatix team.

ENGAGEMENT MODEL OPTIONS - SUGGESTED

RESOURCE AUGMENTATION

In this model, FutureSoft can provide resources for deployment at Zenatix. These resources would be able to execute some/most of the services stated in the offerings section.

This option can kick-off with two resources. Additional resources can be provided on actual need basis.

Resource 1 (Developer) – This resource would have appropriate development skills and would be able to contribute in development activities after little handholding from Zenatix team. Primary focus of this resource would be to understand communication interfaces, generate test data and write modules/scripts for testing.

Resource 2 (Testing & Quality) - This resource would have appropriate testing skills (system & integration testing) and would be able to contribute after little handholding from Zenatix team.

These resources would collaborate together and help in setting up right strategy for testing of platform modules. They are supposed to interact with FutureSoft think tank once a week for guidance towards setting up correct processes.

Zenatix – Testing SOW

SOLUTION CENTER SET UP @FUTURESOFT

In this model, FutureSoft would allocate a space within FutureSoft premise for setting up an offsite solution/development center. A team of 2-3 resources would be allocated for execution of tasks. Tasks to be executed per sprint/month would be agreed mutually.

Zenatix team would empower these resources for execution of tasks. It is expected that within few weeks this team would be ready to deliver various tasks such as,

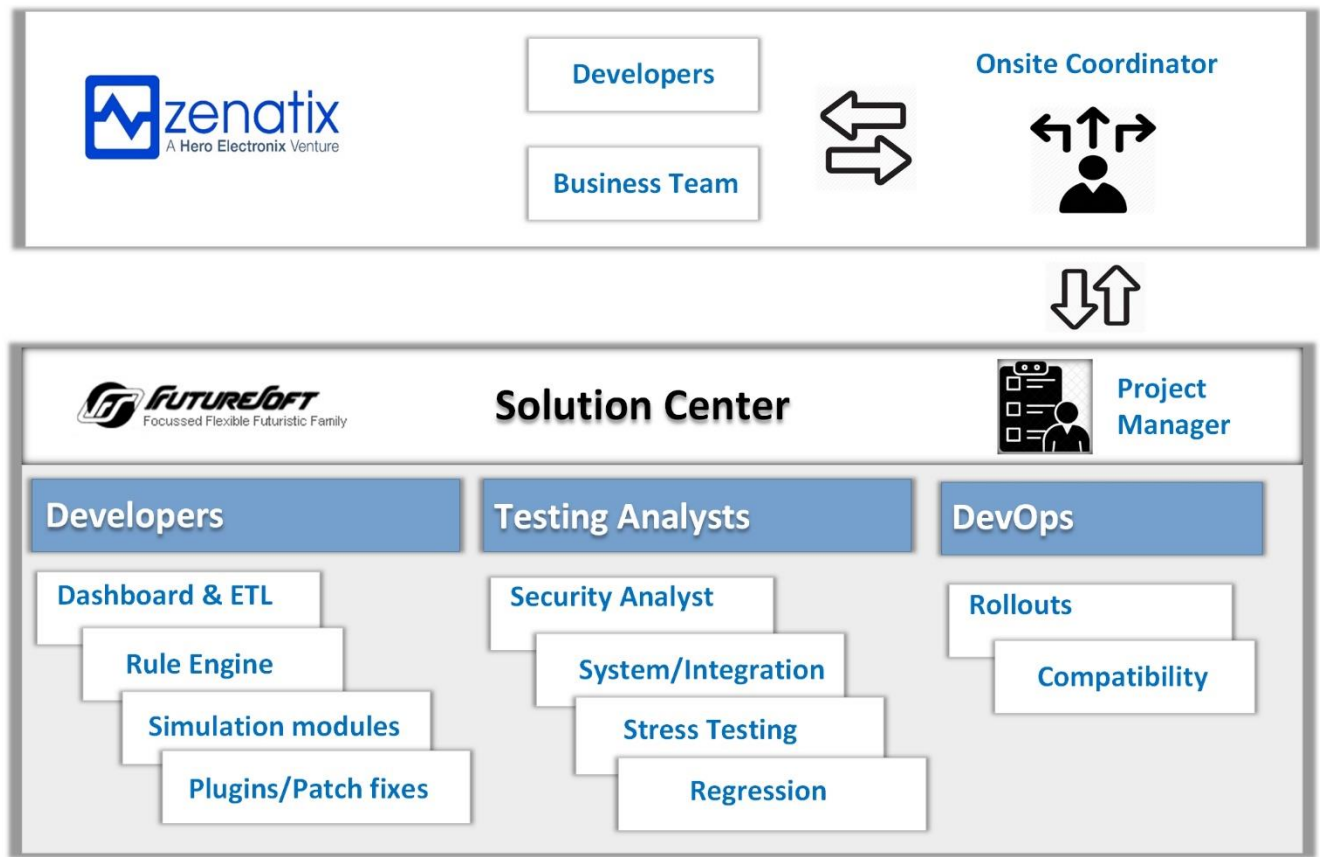
- Implementation of solution at new sites,
- Executing test strategy,
- Develop test modules
- Generate test data for automation testing.

Scope of work for this team can be mutually agreed. Suggested methodology for execution would be Agile.

This model would have advantage of senior people involvement from FutureSoft team to ensure smooth execution and control quality of deliverables.

Onsite/offsite meetings calendar per week for this engagement can be mutually discussed and agreed.

Engagement Model – Solution Center



Zenatix – Testing SOW

COMMERCIALS

RESOURCE AUGMENTATION

Developer Resource – 150K per month

Testing & Quality Resource – 140K per month

Additional resources can be provided on need basis.

SOLUTION CENTER SET UP @FUTURESOFT

Rate – 160K per month per resource

- FutureSoft would allocate senior resources as well on need basis to ensure timely & quality execution of tasks.
- Additional Resources can be added quickly if required.

FINAL ENGAGEMENT MODEL (DISCUSSION DATED – 23 JULY)

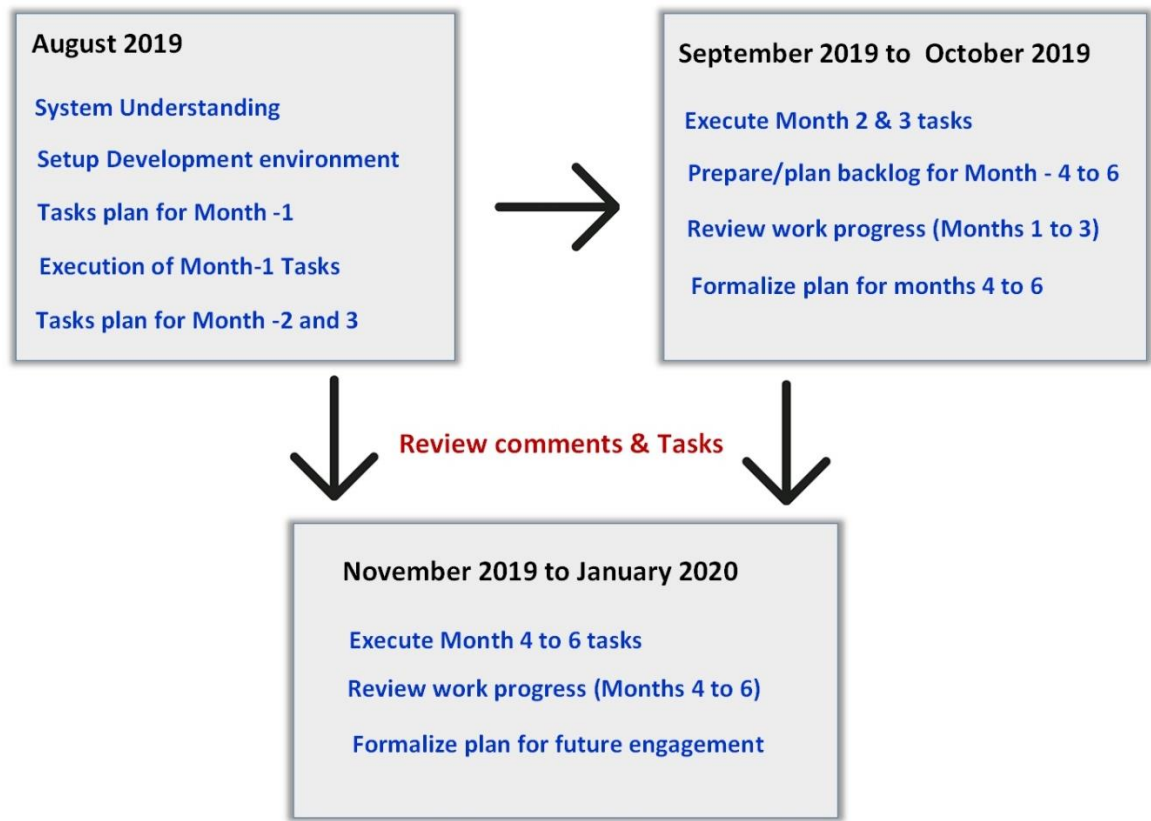
Out of the two suggested engagement model options, ***Solution Center option#2*** has been chosen.

For first 3 months, two resources, i.e. a simulation developer and a tester, would be deployed onsite. In addition, a Senior Architect would be allocated to oversee smooth execution of the July model on once a week basis. Responsibility of these resources during first three months would be to execute tasks related with testing & simulation modules. Rule engine implementation was also discussed. It can be included in the plan of action during 4th and 6th month or later based on mutual agreement.

Plan of first month would be finalized in first week itself. Plan for 2nd and 3rd month would be finalized in 1st month of execution.

Plan for months 4 to 6 would be finalized before end of 3rd month followed by a review meeting between stakeholders.

Special discounted price for the first six months would be 1000K instead of 1920K.



PAYMENT TERMS

- 2,50,000 - Start of engagement (August 2019)
- 2,50,000 - End of month 3 (October 2019)
- 2,50,000 - Start of month 4 (November 2019)
- 2,50,000 - End of month 6 (January 2020)