Department of Information Technology

COURSE CODE: DJS22ITL6015   **DATE:**30-03-2025

COURSE NAME: ISIG Laboratory   **CLASS: T. Y. B.Tech**

NAME: Anish Sharma

Experiment No.6

CO/LO: Describe the types of support that an information system can provide to each functional area of the organization.

AIM / OBJECTIVE: To simulate and compare how evolutionary and revolutionary changes affect user adoption over time using Python.

THEORY:

User Resistance:U ser resistance refers to the opposition or reluctance of users to adopt a new system, process, or technology. It is one of the most common barriers to successful system implementation.

Causes of User Resistance:

Reason	Example
Fear of the unknown	Users unsure of the new system's purpose or operation
Loss of control	Automation reduces users' influence or decision-making power
Increased workload initially	Learning a new system feels harder than sticking to the old
Lack of trust	Users don't believe the system will improve their work

Poor communication	Change not explained properly
Misalignment with user needs	New system does not solve the user's real problems

Impacts on Project:

Department of Information Technology

- Delays in project success
- Reduced system utilization
- Workarounds or refusal to use the system
- Increased cost due to re-training and support

Different Models of User Adoption

a) Rogers' Diffusion of Innovation Model:

Group	% of Population	Characteristics
Innovators	2.5%	Eager, tech enthusiasts
Early Adopters	13.5%	Visionaries, influence others
Early Majority	34%	Deliberate, adopt before average
Late Majority	34%	Skeptical, adopt after majority
Laggards	16%	Highly resistant to change

b) Technology Acceptance Model (TAM):

Adoption depends on:

- **Perceived Usefulness (PU):** "Will this help me?"

- **Perceived Ease of Use (PEOU):** "Is it easy to use?" Higher PU and PEOU increase

the chance of adoption.

c) Unified Theory of Acceptance and Use of Technology (UTAUT):

Factors affecting adoption:

- Performance expectancy
- Effort expectancy
- Social influence

Department of Information Technology

- Facilitating conditions

3. Different Levers for Affecting User Adoption

These are **tools, strategies, or interventions** designed to **reduce resistance** and **increase adoption**:

Lever	Description	Example
Training	Educates users to reduce fear	Workshops, hands-on practice
Incentives	Rewards for early or successful adoption	Bonuses, recognition
Usability Improvements	Makes the system intuitive and userfriendly	User-friendly UI, help guides
Communication	Clear, transparent messaging about changes	Regular updates, newsletters
User Involvement	Involve users in the design process	Surveys, feedback sessions

Support System	Ongoing help and troubleshooting	Helpdesks, super-users
Change Champions	Influential people encouraging adoption	Department heads promoting change

1. Evolutionary Change (Incremental Change / Continuous Improvement)

Evolutionary change refers to a gradual, step-by-step process where improvements or changes are introduced slowly over time. It focuses on refining existing systems, processes, or tools without causing major disruptions to daily operations. Organizations typically adopt evolutionary change when they aim for continuous improvement and prefer minimizing risks. This approach allows users to adapt progressively, reducing resistance because they have time to understand, learn, and adjust to the new changes.

An example of evolutionary change is when a company gradually updates its software modules—starting with one department and moving on to others once the new system stabilizes. Another

Department of Information Technology

example is adding new features to a mobile application one update at a time, allowing users to familiarize themselves with each change.

The benefit of evolutionary change is that it provides a safer way to implement changes. It allows for feedback collection at every step, making it possible to fix problems as they arise. It also lowers the chance of failure because the system isn't drastically altered overnight. However, the drawback is that change might feel slow, and the organization may take longer to realize the full benefits of the new system.

 Examples: • Updating features in an existing software application version by

   version.

- Gradually migrating users from an old system to a new system module-by-module.
- Improving customer service by tweaking processes rather than a full overhaul.

- Introducing one new yoga pose every week in a health app rather than changing the full routine. **Benefits:**

- Reduces the risk of failure.
- Easier for employees/users to accept and adapt.
- Helps sustain long-term change.
- Allows testing and corrections after every step.

Challenges:

- May be too slow for fast-changing industries.
- Benefits might not be immediately visible.
- Continuous change might create change fatigue over time.

2. Revolutionary Change (Transformational / Radical Change)

Revolutionary change, on the other hand, involves sudden, large-scale, and often disruptive transformations. It completely alters the existing system or process in a short period of time. This

Department of Information Technology

type of change is typically driven by external factors such as competition, market demands, technological advancements, or internal factors like leadership decisions aiming for rapid innovation.

Revolutionary change often comes with high risks because it forces users to shift suddenly from their familiar ways of working to something completely new. This can lead to significant resistance as users feel overwhelmed, uncertain, or threatened. However, if managed properly, revolutionary change can bring massive benefits, helping organizations leap forward and gain a competitive edge.

For example, replacing an entire manual workflow system with a fully automated software overnight is revolutionary. It forces everyone in the organization to adapt quickly, changing how

they perform their tasks entirely. Another example is when companies, due to a crisis like a pandemic, switch from office-based work to 100% remote operations in just a few days.

While revolutionary change can lead to rapid gains and show bold leadership, it requires strong planning, effective communication, intensive user training, and support to overcome resistance. The consequences of failure are high because reversing such changes is difficult once the process has started.

Examples:

- Replacing an entire legacy ERP system with a new cloud-based platform overnight.
- Sudden mergers or acquisitions changing entire organizational structures.
- A company switching overnight from office work to full remote work (as in the COVID19 pandemic). • Implementing AI-driven decision systems that replace human judgment.

Benefits:

- Faster implementation of new strategies.
- Potentially greater competitive advantage if executed well.
- Demonstrates strong leadership and innovation.

Department of Information Technology Challenges:

- High resistance from users due to fear of unknown.
- Risk of system failure or adoption failure.
- Requires significant training and change management effort.

OBSERVATION:

Compare Evolutionary change across all user groups compared to revolutionary change import

matplotlib.pyplot as plt

```python
# Data time_periods = range(20) adoption_evolutionary =
[min(100, 5 * t) for t in time_periods] adoption_revolutionary = [0
if t < 10 else 100 for t in time_periods]


# Plotting plt.plot(time_periods, adoption_evolutionary, color='b', linestyle='--',
marker='o') plt.plot(time_periods, adoption_revolutionary, color='r', linestyle='-',
marker='x')


# Graph Labels plt.title("User Adoption Over Time: Evolutionary vs Revolutionary
Change") plt.xlabel("Time Period") plt.ylabel("Adoption Percentage")
plt.legend(["Evolutionary Change", "Revolutionary Change"])  # Using legend for labels
plt.grid(True) plt.show()
```
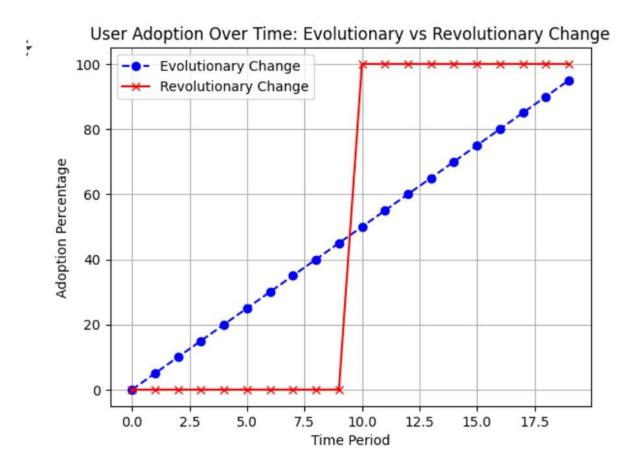
Department of Information Technology

User Adoption Over Time: Evolutionary vs Revolutionary Change

CONCLUSION QUESTIONS:

1. Explain how user resistance affects system adoption.

REFERENCES:

1. changinghighered.com