# GT Journey: The Importance of Accessible Rich Data Sources to Enable Innovation

Matt Sanders[(✉)], Russ Clark, Brian Davidson, and Siva Jayaraman

Georgia Institute of Technology, Institute for People and Technology,
75 5th St NW, Atlanta, GA 30316, USA
{msanders, russ.clark, bdavidson, jsiva}@gatech.edu

**Abstract.** GT Journey (gtjourney.gatech.edu) is an initiative, which empowers members of the Georgia Tech community to develop and deploy applications and services through access to resources (tools, data, services, space) and mentors with technical and domain expertise. The genesis for this initiative comes from a long history of facilitating application and service development for students by students in classroom and entrepreneurial settings. This paper reveals many of the lessons learned from this participatory design, build, and deploy initiative, which may be applicable to a variety of activities in educational, civic, and industry innovation settings.

**Keywords:** Civic computing · Participatory design · Open data · Development communities · Mobile computing · Hackathons · Innovation ecosystems · Devops

## 1 Introduction

GT Journey (gtjourney.gatech.edu) is an initiative, which empowers members of the Georgia Tech community to develop and deploy applications and services through access to resources (tools, data, services, space) and mentors with technical and domain expertise. The genesis for this initiative comes from a long history of facilitating application and service development for students by students in classroom and entrepreneurial settings. This paper reveals many of the lessons learned from this participatory design, build, and deploy initiative, which may be applicable to a variety of activities in educational, civic, and industry innovation settings [1].

The GT Journey initiative began as a collection of established projects which were created to empower members of the Georgia Tech community to create solutions for the campus; this included support for the development of mobile, web, kiosk, and desktop applications and services; creation of tools and platforms for emerging mobile and mixed reality experiences using the Argon Augmented Reality Browser [2, 3], as well as a variety of research projects which relied on operational data from the campus. The initiative leveraged the Georgia Tech Convergence Innovation Competition (CIC cic.gatech.edu) [4] by offering a dedicated Fall installation focused on campus facing solutions and experiences.

GT Journey is a production of the Georgia Tech Research Network Operations Center (RNOC) – which manages relationships with industry, civic, and academic

partners in support of making the campus a living lab. This student led organization acted as the intermediary between student innovators and campus data owners. This included providing development platforms, building restful web services, building and maintaining reference applications, shepherding teams, and staffing a lab space with a wide array of resources.

The GT Journey initiative provided the necessary mandate and resources to move to a more sustainable model in support of student innovation impacting the campus experience. By offering platforms and services over an extended period of time across a diversity of data sets, the project was able to provide authentic live data for all stages of prototyping, implementation, deployment, and extended research. These resources combined with the support from Georgia Tech's President established the trust necessary for unlocking data, breaking silos, creating shared platforms and resources, building partnerships, and affirming the value of incremental design thinking through the campus as a living lab.

## 2   Approach

Our approach for the initiative was based on and informed by over 10 years of supporting student innovation and research with campus data; and in service to courses focused on experiential learning, conducting industry sponsored research, and stewardship of entrepreneurial innovation competitions [4]. In this section we present our observations and the key lessons learned as a result.

### 2.1   Validate Your Perspective

Through our work we have observed a remarkable desire from students and other motivated community members to leverage their course, competition, and independent work to improve campus experiences for themselves and other members of the community. In most cases these individuals are motivated to improve a customer experience, which they find inconvenient or inefficient. Because these individuals don't know the realities associated with the services or processes they want to improve; they often face challenges or make assumptions, which reduce their chances of success significantly. For example a recurring case involves solutions which help select a place to eat based the line length or number of tables available. Invariably the solutions proposed are done without considering that restaurants depend on queues and turnover to be viable enterprises, and therefore have a negative incentive to report this type of information, or allow it to otherwise be collected and published in real time. This lack of understanding of the realities of the service or process they wish to improve can cripple a project from the start.

*Lesson 1: There is a need to coach or connect innovators so that their solutions take into account the Point of View of the different players interacting with a service or process they intend to improve, leverage, or disrupt.*

Consistent with design thinking [5–8]; in our work we challenge innovators to think beyond their own needs and experiences by facilitating meetings with the

operators of a service (ex: restaurant owner, bus operator, transportation manager) and surveying other types of users of a service so that they can better understand differing goals, needs, and constraints of the service.

## 2.2  Escape Groundhog Day

As anyone who has attended a hackathon, taught a project based course, or those in the venture capital business can tell you - there are no new ideas. There are only variations on old ideas which have either failed or which can be done differently through a new approach and with better execution. In many of our project based courses the lack of access to real data significantly contributes to execution challenges. After a few iterations of our Mobile Applications and Services course we experienced this "Groundhog day" type experience where the faces had changed, but the problems identified, the solutions proposed, and the amount of progress made was nearly the same as in previous semesters.

*Lesson 2: Without access to real data, application quality will suffer and the chances for the work to have an impact are greatly reduced. The key to fostering the creation of novel and useful applications is in making rich data sets available to developers.*

Over time we created platforms, which allowed students access to live campus data, required students to contribute to this platform by publishing their applications and any derivative data services they created.

## 2.3  Respect Data Diversity

A common mistake is to assume that because data exists, is accessible for your purposes. Among the many technical issues we have experienced unlocking campus data for use by student developers we find the following situations most prevalent:

*Data* Silos- data exists but is locked in a proprietary database.

*No Context* - the way that data is stored for its primary function is not compatible for the new use cases or services. This includes limits on data resolution or quantity.

*Limited Systems* - the systems where data are stored are not real-time or are not intended to be accessed by as many clients or at the transaction rate required by the new use case or service.

*Multiple Systems* - the data you want is spread over many systems which could have inconsistent ways of referencing the same items.

*Lesson 3: Just because you can imagine or have seen data of the form necessary for an application doesn't mean that data can easily be curated or maintained.*

The solutions to these challenges are varied, but they all start by developing a firm understanding of the situation and looking for sustainable opportunities to make data available. In some cases we were not able to gain access to data locked in legacy and/or proprietary systems and chose to fail fast; however, in most cases we were able to gain access to imperfect data sources and develop appropriate ways for developers to access the data without putting the primary function at risk. This included enforcing access

controls on the data in some cases, as well as providing a separate service interface for application developers which would protect the primary function from rogue or aggressive activities. In other cases we worked closely with the data owner to improve upon their data collection or data services. A good example of this is our facilities data; where we were able to give access to fine grained data across multiple systems which was not possible with the collection of proprietary function-specific tools. Another good example of this was the instrumentation of bus location data collection where we were able to provide a commercial location prediction service access to our data while retaining this data for our own applications and services as well as research projects which improved the overall operation of the bus system [9, 10]. Our campus map efforts are another example of this where no fewer than 3 groups maintained 5 different databases of campus locations; each with a different set of fields, and without a common index. We were able to partner with these groups to develop a single database which could scale to everyone's needs, contained the diversity of information required by all applications, was curated and tied to authoritative campus map information. In all cases our success in gaining access to data and making it available was because we were opportunistic and circumspect in choosing targets and recognized when we could amortize our efforts, and provide value to a good partner.

## 2.4   Bring an Olive Branch

Beyond the technical challenges discussed above other barriers exist in the path to accessing data and evolving or leveraging data from existing services. Here are some we have experienced in our work:

**Policy.** privacy, risk mitigation, and/or security policies which are narrowly interpreted or crafted only for known use cases can make it difficult for data and service owners to allow access to data. In general data stewardship policies and procedures are defined around protecting data and institutional access as opposed to openly sharing and encouraging broad use by all members of the community [11]. In cases where there is ambiguity around making data accessible the process slows considerably.

**Loss of Control.** Even requesting access to non sensitive data can reveal fears of lack of control in what might be done or revealed by the data. This loss of editorial control (ex: defining the user interface or not allowing certain analysis) can lead decision makers to resist transparency [12]. This perceived loss of control can also be projected into the future, where the service owner sees innovation by others or supporting external access to data as limiting their opportunity to provide value (relevancy) or ability to make necessary changes to their services.

**Perceived Value.** If the perceived value relative to the work and risks is limited, then it is difficult to justify expending resources or extending risk. Whether it is adding a link to a web page, doing simple data entry or curation, or supporting access to data during a migration, upgrade or transition of the service; even these seemingly small amounts of work are often difficult to justify or commit to when resources are sparse or overcommitted. This is especially true for well established (and funded) services where

existing service offerings are perceived to meet the needs of the community, and adding the complexity of engaging with outsiders only adds risk.

As with the data access barriers discussed above, we discovered a variety of solutions; though most centered around two lessons:

*Lesson 4: without trust and value there is no basis for sharing data with developers; addressing sustainability is a key to establishing this trust with data owners.*

We discuss the basis for sustainability in the next section in more detail. Our approach to these challenges in general required respect for the data stewards regardless of the barriers. While we pushed (often passionately) for access to data for campus constituents, we could not dismiss their concerns outright and worked hard to address them. This in the end informed our interactions with developers and influenced the services we provided in meaningful ways. While it is certainly possible for individual development teams to establish trust with service owners, this one-on one model is not scalable across a variety of services and applications; and more importantly it is not sustainable given the limited duration of most projects and/or engagement by members of the campus community. Our approach required that we act as intermediaries who could manage the relationships with data and service owners, support the resources used by application developers and users, and assist in transitioning applications to production or new development teams.

Because we were not focused on one team or application and had a longer timeline, we were able to build trust by accepting data in whatever form we could get it, with or without updates, and by leveraging these successes based on small initiatives, we could eventually build up to more sensitive and/or challenging data. It is worth noting that a significant number of applications and services require read-only access to data. By providing developers and application users with access to a cached read only (but updated) repository of data we were able to hurdle the barriers above in order to free data. This approach worked well with both our learning management system and our building information system, where the challenges in accessing data and/or the risks of allowing developers to write data back into the system were insurmountable and would otherwise kept this data inaccessible.

Another way we were able to build trust is by developing new & novel use cases which were not in competition with the core service but which gave the data provider some credit and association with a new technology developed by students; creating a halo effect. This was certainly the case with mixed reality applications and services [2, 3]; where the data owner didn't have the necessary resources or expertise to develop to these emerging technologies.

## 2.5   Sustainability Throughout

**Building Strong Communities.** Maintaining quality relationships with data and service owners was essential in order to maintain trust and show value, but also to provide opportunities to expand access to new data sources, discover new use cases, and to learn about transitions of services or processes. In some cases we also developed reference applications, which were of operational value to the data owner. This was the

case with our utility data where we were able to provide a simple reporting capability which was previously not possible in their soloed and proprietary environment.

By far our most extensive engagement activities were focused on Developers and Designers. This was done through a diversity of efforts beginning with outreach activities in a variety of primarily project based courses and other student facing programs (Senior design and capstone projects, hackathons, and student organizations focused on entrepreneurship, application development, and innovation). We also staffed and outfitted a lab where we made development resources available to students, held Tutorials and advising sessions, and ran Competitions [4] and hackathons to encourage development in particular areas or targeting specific platforms. All of these engagement efforts were responsible for building Community - an essential element for sustaining development of the platforms and applications. These efforts relied on a large number of paid and unpaid student research assistants to conduct outreach and training, shepherd teams, maintain platforms, check-in/out resources, and in general become the domain experts acting as intermediaries between the data and service owners and the student development teams. In addition a few full time staff are employed to support this pipeline of student workers, essential in maintaining consistency as well as providing the technical and institutional oversight necessary given the data we are trusted with and the scale of operations.

*Lesson 5: By placing (senior) student staff in leadership positions, we benefit from experiential learning, and consistently attract high caliber students who are highly motivated to take ownership of the applications, platforms, and lab culture. By pipelining this support we are constantly refreshing and adjusting the skillsets provided and maintaining a relevant point of view of student life.*

**Platforms Build Trust.** We have developed and maintain platforms, which are designed to address the concerns of partners providing data or services. We are able to avoid student developed stove-pipes by requiring students to develop applications with a strict separation of front-end from back-end; such that the back-end data services are not specific to any one application or platform. We also require any new, derivative, or application specific services developed to be implemented as RESTfull web services [13] and hosted on our platform (GTdevHub). This platform provides access to data from a variety of service owners and previous student contributors, and now services unique data sets available to student developers and researchers.

*Lesson 6 : Enforcing meaningful standard methods for data access and separation of back-end services from front-end applications supports sustainability.*

This methodology has shown it's validity and value numerous times over the years. Perhaps the best example is our bus location API which was originally developed to support a 3rd party developer paid to provide bus arrival predictions. It was then used by dozens of student development teams in classes and competitions for a variety of mapping and tracking applications and became the back-end for a mobile web application which provides a real-time map of bus locations along with predicted arrival times. It was also used for a kiosk based web application and a Google glass live tracking application, and became the basis for a research project which was transitioned to a production service that prevents bus bunching along the route. While the back-end service has evolved over the years, it has not fractured. This not only makes sustaining

the back-end code easier, it also means that the API is available for future application platforms we might not conceive of today.

**Services Rise Tides.** We provide three core services give application developers a trusted method for writing applications, which allow end users access to their personal information provided by data owners. One service brokers authentication and authorization in a consistent way for applications and services developed on our platforms. Another provides a cache of data where the authoritative source for the data requires some translation or cannot provide direct access to data by untrusted applications. Add to this a service, which leverages end user authentication and adds application specific access controls for individual users, which are not available in the native interfaces to data sources.

*Lesson 7 : providing consistent and trusted services aided in overcoming the challenges data owners have with making data available to developers, served to "rise the tide" for developers, and made the applications more viable to end users alike.*

In this model, we are establishing trust between our servers and the data source and then we are responsible for ensuring that access is only granted to those who should have access. This delegated responsibility is another example of the need for an intermediate referenced in the previous section.

**Applications Need Homes.** As discussed in Sect. 2.2, the churn of projects and students without a means of one team's experience informing a future team's efforts is a significant challenge for those looking to capitalize on engaging student innovators.
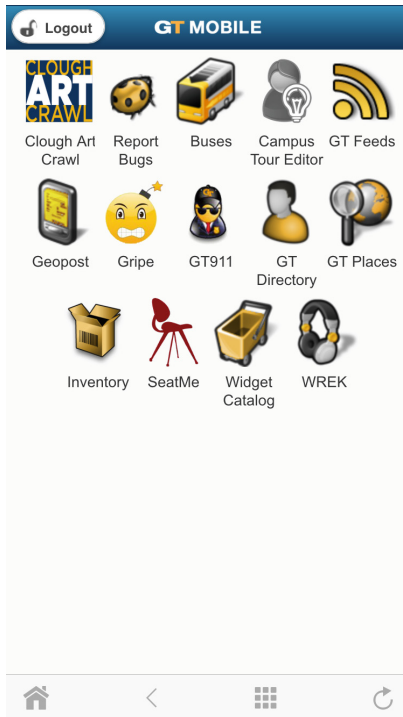


**Fig. 1.** GT mobile

*Lesson 8 : by creating common application platforms we were able to retain the student work product in order to continue or learn from the work, but also provided valuable services to developers by hosting their applications on a robust and always accessible server infrastructure and facilitating immediate and iterative testing by themselves and other campus users with live data.*

These attributes provided incentive for use of the platform (GT mobile) shown in Fig. 1 below, which in turn drove adoption and therefore helped with retention of the work for use by future students to improve, expand, or at least learn from the effort.

**Leverage Hybrid Vigor.** While our platforms had succeeded in establishing trust with data owners and provided value to developers, we continued to see the same ideas developed repeatedly. In many cases the applications addressed relevant needs and were sound prototypes; but were not feature complete and required more refinement than was possible in the context of a class, hackathon, or competition.

*Lesson 9 : even with data, resources, support and platforms - to fully develop even simple applications such that they can be deployed requires significant effort and refinement.*

Mobile real-time bus tracking is an example for which it was relatively straight forward to conceptualize the application and the architecture at a high level. However the user experience design, including responsiveness, visual abstractions, and supporting front end application architecture required multiple iterations and significant
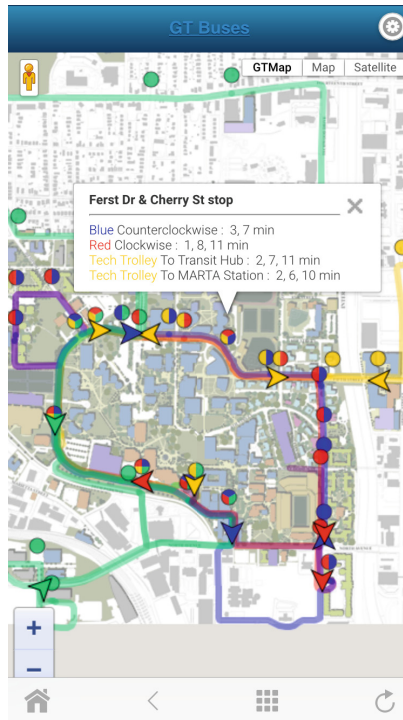


**Fig. 2.** GT busses

work from those with a variety of skillsets. After seeing dozens of attempts, we took it upon ourselves to develop a reference application shown in Fig. 2 above, which was informed by the best ideas we had seen. The "busses" application is by far our most popular application on the platform among users, and has also provided numerous developers a needed starting point for understanding how to work with the busses API. Following this, we built a suite of reference applications were built based on past student projects, including a campus directory for people, a building directory and campus map, among others.

## 3   Discussion and Conclusion

The GT Journey initiative has been successful in fostering and sustaining a community around the development of applications and service improvements for the campus. The lessons learned presented in this paper provide generalized guidance relevant to those undertaking engaging data owners and disjoint or ephemeral development and design teams in the educational, civic, and industry innovation settings. In each setting the degree to which these lessons hold will depend largely on understanding and embracing the tension between producing innovative solutions and the desire for production ready services. We have benefited greatly from both the desire to provide production services, and the acceptance of "good enough" applications. This has allowed for a DevOps [14] culture where the momentum behind further development is proportional to the value perceived or demonstrated. Indeed the vast majority of applications fail or whither in this environment, but the risk and cost of supporting those failures by data and services owners is small relative to the value. By allowing the community to innovate the applications they want service owners have the opportunity to leverage innovation to gain new services in the best case, and key insights in the worse case.

Note that we did not discuss top down approaches to the barriers above. While the GT Journey Initiative was sponsored by Georgia Tech President Bud Peterson, and funded in support of the Georgia Tech strategic plan; we only leveraged this to make introductions and validate the project. Beyond this we firmly believed that a bottom up approach among stakeholders was the only way to ensure functional trust.

It is reasonable to expect that participatory and shared models will become more necessary as a number of areas of interest today mature; these including Internet of Things, Cloud and DevOps [14], and Smart Cities. In each of these there is an underlying assumption of or desire for shared services or resources and leveraging one thing for many purposes. Designing future-proof applications, architectures, and services in such a world will not be possible. Instead the need all parts of the system to be adapting to the changing needs of the rest of the system will require approaches like those presented in this paper in order to be successful.

# References

1. Gandy, M., Baird, L.D., Levy, L.M., Lambeth, A., Mynatt, E., Clark, R., Sanders, M.: Midtown buzz: bridging the gap between concepts and impact in a civic computing initiative. In: M. Kurosu (ed.) Human-Computer Interaction, Part III, HCII 2015, LNCS 9171, pp. x-y. Springer, Heidelberg (2015)
2. MacIntyre, B., Hill, A., Rouzati, H., Gandy, M., Davidson, B.: The argon AR web browser and standards-based AR application environment. In: 10th IEEE International Symposium on Mixed and Augmented Reality (ISMAR), pp. 65–74 (2011)
3. Spieginer, G., MacIntyre, B., Bolter, J., Gandy, M., Lambeth, A., Levy, L., Baird, L., Mynatt, E., Clark, R., Sanders, M.: The evolution of the argon web framework through it's use authoring community-based mixed reality applications. In: Human Computer Interaction International, Los Angeles, CA (2015)
4. Clark, R., Sanders, M., Davidson, B., Jayaraman, S., DiSalvo, C.: The convergence innovation competition: helping students create innovative products and experiences via technical and business mentorship. In: M. Kurosu (ed.) Human-Computer Interaction, Part III, HCII 2015, LNCS 9171, pp. x-y. Springer, Heidelberg (2015)
5. Dix, A. (ed.): Human-Computer Interaction, 3rd edn. Pearson/Prentice-Hall, New York (2004)
6. Stickdorn, M., Schneider, J.: This is Service Design Thinking Basics, Tools, Cases. BIS Publishers, Amsterdam, The Netherlands (2011)
7. Ylipulli, J., Luusua, A., Dey, A.K.: Urban computing in theory and practice: towards a transdisciplinary approach. In: Proceedings of the 8th Nordic Conference on Human-Computer Interaction: Fun, Fast, Foundational, New York, NY, USA, pp. 658–667 (2014)
8. Bisker, S., Gross, M., Carter, D., Paulos, E., Kuznetsov, S.: Personal, public: using DIY to explore citizen-led efforts in urban computing. In: CHI 2010 Extended Abstracts on Human Factors in Computing Systems, New York, NY, USA, pp. 3547–3552 (2010)
9. Bartholdi III., J.J., Clark, R.J., Williamson, D.W., Eisenstein, D.D., Platzman, L.K.: Building a self-organizing urban bus route, pp. 66–70 (2012)
10. Christopher, B.: Bartholdi's NextBuzz Receives Innovation Award
11. Socrata, 2014 Open Data Benchmark report
12. Lathrop, D., Ruma, L.: Open Government: [Collaboration, Transparency, and Participation in Practice], 1st edn. O'Reilly; [Mass.], Beijing ; Cambridge (2007)
13. Richardson, L.: RESTful web services. O'Reilly, Farnham (2007)
14. Hüttermann, M.: DevOps for developers. Springer, New York (2012)