# Team Project Deliverables

You will form a team of 5 based on the Lab section you are in. If the section has less than 20 people enrolled you will form teams of 4 people. There are 5 positions that you will rotate through in development sprints. Each person will take a primary role—but that role is not exclusive. Each team, starting week 1 in sprint 1, will outline a project plan of goals they aspire to accomplish. At the end of the sprint the team will present their project to the entire class. See below for written paper requirements:

The outputs will be demonstrated through your application code and infrastructure code on GitHub, your project progress will be tracked using a Project Management tool and Slack will be used to show code commits and team discussion. Bugs will be tracked and resolved on GitHub Issues. Operations will be tested to show your Visio diagrams match your output and how automated your build infrastructure is.

## Team Roles

* Project Manager responsibilities include but are not exclusive to:
  + Responsible for managing resources, team, goal setting, and achieving set goals for that 3-week sprint
  + Responsible for determining resource blockers and re-allocating resources
  + Responsible for preparing reports and demonstrating application build
  + Responsible for assuring working product and documenting necessary build instructions
  + Responsible for allocating issues and issues tracking on GitHub
* Developer responsibilities include but are not exclusive to:
  + Responsible for coding and programming
  + Responsible for showing consistent code commitment and deployment based on the language the team selected
* Jr. Developer
  + Responsible for working with the Developer in coding tasks
  + Responsible for SecOps (Security Operations) in testing/searching code for security vulnerabilities
  + Responsible for working with UI/UX developer to resolve user bugs
* IT Operations responsibilities include but are not exclusive to:
  + Responsible for ensuring and deploying infrastructure and code
  + Responsible for ensuring all team members can deploy all code locally and to production
  + Responsible for documenting and drawing all infrastructure
* UI/UX Developer
  + Responsible for designing and documenting the UI/UX of the project
  + Responsible for justifying the UI/UX decision
  + Responsible for testing responsive design
  + Responsible for user testing and filing of bugs on GitHub Issues

## Project Base Requirements

Each Project will have additional specific details to be delivered once the exact project subject is determined. But in this case we are going to disallow non-frameworked PHP and Google Firebase based applications. Note some of the requirements below span across multiple categories.

### Coding and Programming

* Language and framework of your choice (ITM 311, ITM 312, ITMD 361 & 362, ITMD 411) Suggestions:
  + Android
  + PHP – Cake/Symphony/Zend
  + Java – Spring/JSP/Hibernate
  + JavaScript – NodeJS/Angular/React/Vue.js
  + Ruby – Ruby on Rails
  + C# - ASP.NET

### Infrastructure and IT

* Operating System Platform of Choice (ITM 301, ITMO 356)
* Use of Data Store (ITMD 321, ITMD 411, ITMO 356)
  + Database or similar storage technology
* Datastore makes use of master/slave replication (~ITMD 321, ITMD 411)
  + Master for database writes
  + Slave for database reads
* Creation of Data Schema
* Creation of infrastructure diagram tool and work flow (Visio or comparable) (ITM 301, ITMO 356)
* Team must generate at least 15 real “test” users and proper data to test functionality of a system
  + No system is ever used “blank” always fill it up with real data.

### Security

* Data encrypted at rest (ITMS 448)
* Use of https (ITMS 448, ITMO 356)
  + Self-signed certs
* Login authentication mechanism. Google authentication for login, such as:
  + [SQRL](https://www.grc.com/sqrl/sqrl.htm)
  + [Google OAuth2](https://developers.google.com/identity/protocols/OAuth2)

### UI/UX

* Use of Responsive Design (where applicable) (ITMD 361, ITMD 362)
* Use of user authentication (ITMD 411)
  + Must use HTTP Session
  + Different UI for Unauthenticated users
  + Have read/only features for unauthenticated users
  + Different UI for Authenticated users
  + Have a user account management page (EDIT Page)
  + Different UI for Administrative users
  + Have features for Operations in application
    - database dump
    - database restore feature
    - turn any/an upload feature to read-only
  + UI is modified per authenticated user via CSS
* Layout design (ITMD 361, 362, ITMM 471)
  + Diagrams of site functionality using layout tool
  + Diagrams of colors, fonts, and other usability features

### Project Management

* Management of project progress (ITMM 471)
  + Trello (not provided) or JIRA (account will be provided)
  + Slack
  + GitHub
  + GitHub Issues to resolve bug posts from UI/UX tester
* User story - how the user will experience the site

## IT Operations

* Application needs to include:
  + A load balancer
  + minimum of 2 front-ends
  + A memory caching layer (such as Redis or Memcached)
* Database needs to have 15 users with actual data contained at deploy time
  + Needs to have Master Slave replication and or 3 nodes for replication
* Creation of Dev Environment on local laptop (ITMO 356) (ITM 301)
  + Application must always be in a working state
  + Each team member must be able to deploy the entire environment via script on their own hardware
  + Environment must be configurable via a scriptable deploy
  + No manual editing or installing
    - Note - Test and Production environments will be built after sprint 03

## Written Report Content

At the start of each sprint each person will contribute their deliverables. This list of atomic events will be used to grade each person individually for each sprint. The project manager will be responsible to collecting the team’s input and prepare a report. You will receive feedback from me on this report. The report should state the following at the minimum with additional information will be released per sprint.

* List the team members and describe their functions during the spring
* Describe the groups stated (atomic) goals for the sprint
  + Describe how each of these goals were accomplished
  + Explain and show how bugs were found, tracked, and delegated
  + Explain any security assumptions made and explain how these were discovered and mitigated
* List the goals for the next sprint
* Each team member should submit a small comment on their own accomplishments and describe what they did and explain any decisions referencing any lectures or material from the text book or web
* Reference decisions from any of the videos, guest lectures, and class presentations, or text book

## Deliverables

Each team will have different requirements but there are core requirements for all.

* Visio or other diagraming tool for the outline of your application
* Create a folder named: diagrams in your team GitHub account.
* Create UI/UX layout designs of every page – including content layout, colors, fonts, etc. etc.
* Place these in your diagrams folder
* ReadMd.md must have each team members name and contact email
* Must include build instructions to automatically build the application from the command line including all necessary dependencies, instructions, and assumptions

## Final project Report

1. Create and summarize into a final report detailing the accomplishments of the 6 sprint reports.
2. Explain the overall accomplishments of the team based on the content of the 6 reports
3. Analyze the completeness of the project in fulfilling the stated goals (self-assessment, how close did you get?)
4. Explain in detail from the textbook and resources – some features that were not implemented are ok if you can explain accurately why that happened.
5. Explain your technology and stack choices
6. Include your team’s backgrounds