

APAD Phase 1

Hannah Clark
Jeff Wang
Joel Brisola
Yoav Ilan
Martin Dieck

R1-1: Project Plan

Iteration Length: 1 week

Sprint velocity:

Sprint 1: 20 points

Sprint 2: 16 points

Sprint 3: 12 points

Collaboration tools used: Github, Microsoft Azure DevOps, WhatsApp, Zoom

Implementation methodology: Agile Scrum Development

Stakeholder Needs:

SN0: Accepted quality and reliability metrics

SN1: Create and maintain secure user accounts and projects on the system

SN2: View the status of all hardware resources in the system

SN3: Request available hardware resources

SN4: Checkout and manage these resources

SN5: Check-in the resources and get status of all hardware resources in the system

SN6: Deliver PoC within schedule constraints, with support for scalability

System Requirements	Stakeholder needs met	Notes
SR1: PoC shall be delivered within budget and schedule constraint, with periodic updates to stakeholders	SN6	Weekly updates in sprint retros, budget will be maintained at \$0 using Firebase (free service)
SR2: PoC App shall have a front-end web application that allows users to enter inputs and views outputs	ALL	Store inputs and outputs in Firestore database to maintain state.

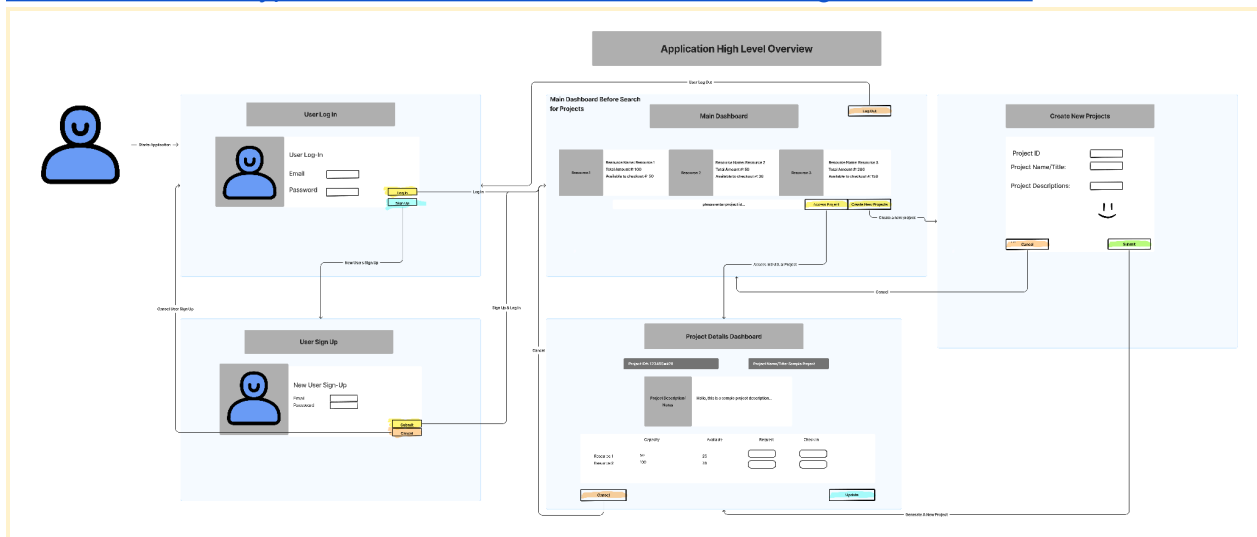
SR3: PoC App shall have a mechanism for encrypting user-id and password	SN1	App will use either Python or Firebase (TBD) encryption for security
SR4: PoC App shall have a mechanism for creating new, unique projects or accessing existing projects	SN1	Projects will not be tied to users (?)
SR5: PoC App shall have a NoSQL database for maintaining user login credentials, project codes, project details, resource details	SN2, SN3, SN4, SN5	Firebase will be used to store values.
SR6: PoC App shall have a system dashboard view that shows the current availability status of resources	SN2, SN5	A brief resource availability overview dashboard should display in the landing page
SR7: PoC App shall have a view for users to access and check in/ out the resources	SN3, SN4	Database will be updated accordingly.
SR8: PoC App shall have robust error-checking to make sure requested # of resources don't exceed what's currently available	SN3, SN4	If the check fails, the user will not be allowed to check out the requested number of resources and will receive an error message.
SR9: PoC app will have robust error-checking to prevent prevent checking in more resources than are in a project	SN5	If the check fails, the user will not be allowed to check in the resources and will receive an error message.
SR10: PoC app shall have a reliable server	SN0	Usage reports will be generated from firebase (App's server host) and firestore (App's

host and database, and will show usage report on the host side to demonstrate application's reliabilities and qualities		database)
SR11: PoC app shall have a project dashboard view that shows the currently held resources.	SN4	Users will be able to see all available resources + check out and return resources via this page.
SR12: PoC app shall give users the capacity to create a new user or sign-in with existing credentials.	SN1	User management

R1-2: All Features on Board
Will Be Shared

R1-3: High Level Sketch

<https://www.figma.com/file/kZmDfmIwXImw1vvYJ0twQL/APAD-Summer-Project-Application-High-Level-Overview?type=whiteboard&node-id=0%3A1&t=8tIMPg3PEMt1YS2c-1>



R1-4: Choice of Tools and Approach

We have a react front end interface with a python backend; however, we are still deciding if we want to use Firebase or MongoDB for our database system. We utilized Github, Microsoft Azure DevOps, and VS Code to help us with the development process. Additionally, we used WhatsApp and Zoom to communicate for stand ups throughout the week.

