CS2310 Software Engineering
Fall 2015
Project Report

Displanner
(Distributed Scrum/Sprint Planner)

By:
Bhavin Modi
Jose Michael Joseph
Vivek Punjabi

Introduction:

The software industry has evolved dramatically during the recent years. With the recent advances in innovation and ever-evolving software technology, the software demands of consumers have become more complex than ever. The current software development approaches such as traditional waterfall model and prototyping model face many challenges during execution. The teams are usually distributed across the cities and even across the globe. With different time zones and languages, it becomes far more difficult to communicate and even keep track of product updates. The traditional models have no such flexibility to incorporate this ever evolving software industry. To provide a platform to tackle such challenges, software professionals came up with Agile methodology.

Agile methodology is an alternative to traditional project management, typically used in software development. It helps teams respond to unpredictability through incremental, iterative work cadences, known as sprints [1]. Agile development methodology provides opportunities to assess the direction of a project throughout the development lifecycle. The results of this "inspect-and-adapt" approach to development greatly reduce both development costs and time to market.

Scrum is the most popular way of introducing Agility due to its simplicity and flexibility. Scrum emphasizes empirical feedback, team self management, and striving to build properly tested product increments within short iterations.

One of our primary motivations was to allow teams working on mobile locations to be able to use the app to communicate their SCRUM details effectively

Scrum has three basic roles: Scrum master, Product owner and Team. Each of the roles have specific set responsibilities and authorities. To cater each of these different roles on a single standalone application, we propose a design to integrate all these roles in one Mobile application that can be accessed through any smartphone with minimal hardware and software requirements. We have three major components each corresponding to a particular set of users that will use them.

Scrum Manager:

This component of the app is used to enable the user to monitor the developers under his chain of authority as well as monitor other scrum masters.

It has an integrated display which populates the result for both Developers and Scrum Masters and differentiates between the two using the 'MASTER' tag.

It also allows clear demarcation of data ensuring that a developer never has access to SCRUM data outside his group.

The primary objective of the SCRUM master application would be to enable the scrum masters to monitor their developers and other scrum masters with ease.

This ease is especially essential when handling distributed scrum teams which is the premise of this application. In a distributed team one can expect to have members from different parts of the globe coordinating to complete a particular task. In such scenarios it is especially essential to have great communication between the various team members. Breakdown of communication in any form can lead to a mismanagement of the project which would lead to a waste of resources and time. Since one of agile's primary benefits is a short development time, it is critical to have an excellent communication between the various members of the group.

Scrum Developer:

This component of the app is used to enable the developer to update his SCRUM fields as well as keep a track of others on the same team.

Thus if a team member faces difficulty then the developer can quickly respond to them as the information is globally synced.

It also allows the developer to see if anyone else is working on the same problems as they are thus enabling the developer to team up with others to work towards similar goals.

The primary goal of this component of the app was to allow developers to have seamless communication with each other. This helps in a way such that if one individual is stuck on a particular segment of the code then another can reach out to them to help them. Such a collaborative effort would reduce the development time of the entire application and the strengths of each individual can be used to supplement the weaknesses of the other.

Thus our app holds true to the spirit of agile and scrum and enables us to help the team deliver great projects in the limited amount of time available.

This component accesses the ScrumMasters table in the online server to populate its result. By ensuring that scrum master information is on a separate table as compared to developer information we ensure that the security and privacy features of this app are not compromised. By following this model we ensure that developers can never have access to data regarding scrum masters. This is especially vital in the current age where we see an increase in information leaks due to unauthorized people having access to data that they are generally not supposed to have.

Sprint Planner:

Sprint planning is a collaborative effort involving [2]:
1. Scrum Master – to facilitate the meeting
2. Product Owner – to clarify the details of the product backlog items and their respective acceptance criteria
3. Entire Agile Team –to define the work and effort necessary to meet their commitment to complete product backlog items

This component works on a lower level as compared to the Scrum Master and Developer. This is because it deals with tasks which are more fine grained and specific to a single requirement of the product. It tells what is expected from each developer in terms of product requirements. This especially helps when there are bugs related to some particular requirement. It will be ideal to assign it to the developer who was responsible for its related task in the first place.

The 'Sprint Planner' tab is currently visible only to the master. It allows him to create and tasks and bugs for the current sprint and even review the same for older sprints. He can also assign tasks/bugs and any developer in his team or any available developer from another team.

Each task is defined as a Task Card that represents all the details for that particular task. This includes the ID, type (task or bug), name of task, detailed description of the task, priority (high or low), owner (developer responsible for to complete the task) and the status. This task list belongs to the product team of the manager who is currently logged in.

Fig. Task Card representing a task

This transparency is provided by the shared database, which is hosted on a web server. The table responsible for the tasks list has the following columns.

#	Name	Type	Null	Extra
1	id	int(11)	No	Primary Key Auto-Increment
2	type	varchar(15)	No	
3	name	varchar(100)	No	
4	desc	varchar(1000)	No	
5	priority	int(11)	No	
6	status	varchar(15)	No	
7	owner	varchar(50)	No	Foreign Key to 'email' IN Developers
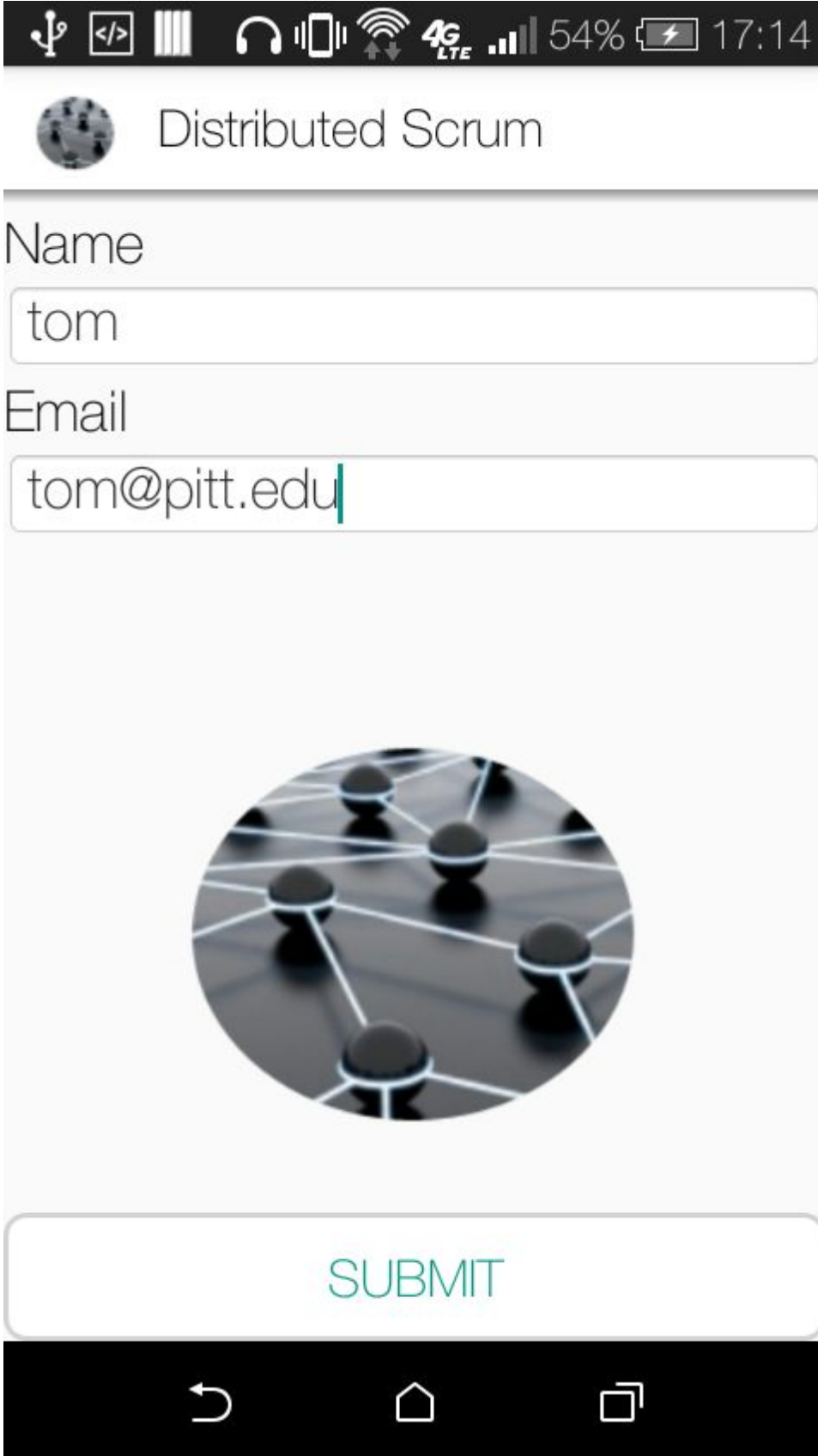
Fig. TaskList table

It accesses the Developers table to access the groups for each of them and display the corresponding developers to assign the task to. The manager can also switch the group from the icon on top and assign tasks to available developers from other groups.

The server also hosts a summary page for the components that display the contents of the table to make it accessible from desktop machines as well.

Scenario:

We first see that Tom, a relatively new Scrum Master, starts up the application. He logs in with this name and email ID.

Notice that as soon as the app logs him in it sets its group as -1. This is the default for all new users. It also provides space for Tom to enter his goals and obstacles.

Now Tom clicks on the gear on the top left corner (the settings button) and changes his group tp 1.

Distributed Scrum

Name

TOM

Email

tom@pitt.edu

Group

1

SUBMIT

Tom now browses other Scrum Masters' information to see what they have been up to. Tom can browse both Scrum Master as well as Developer information.
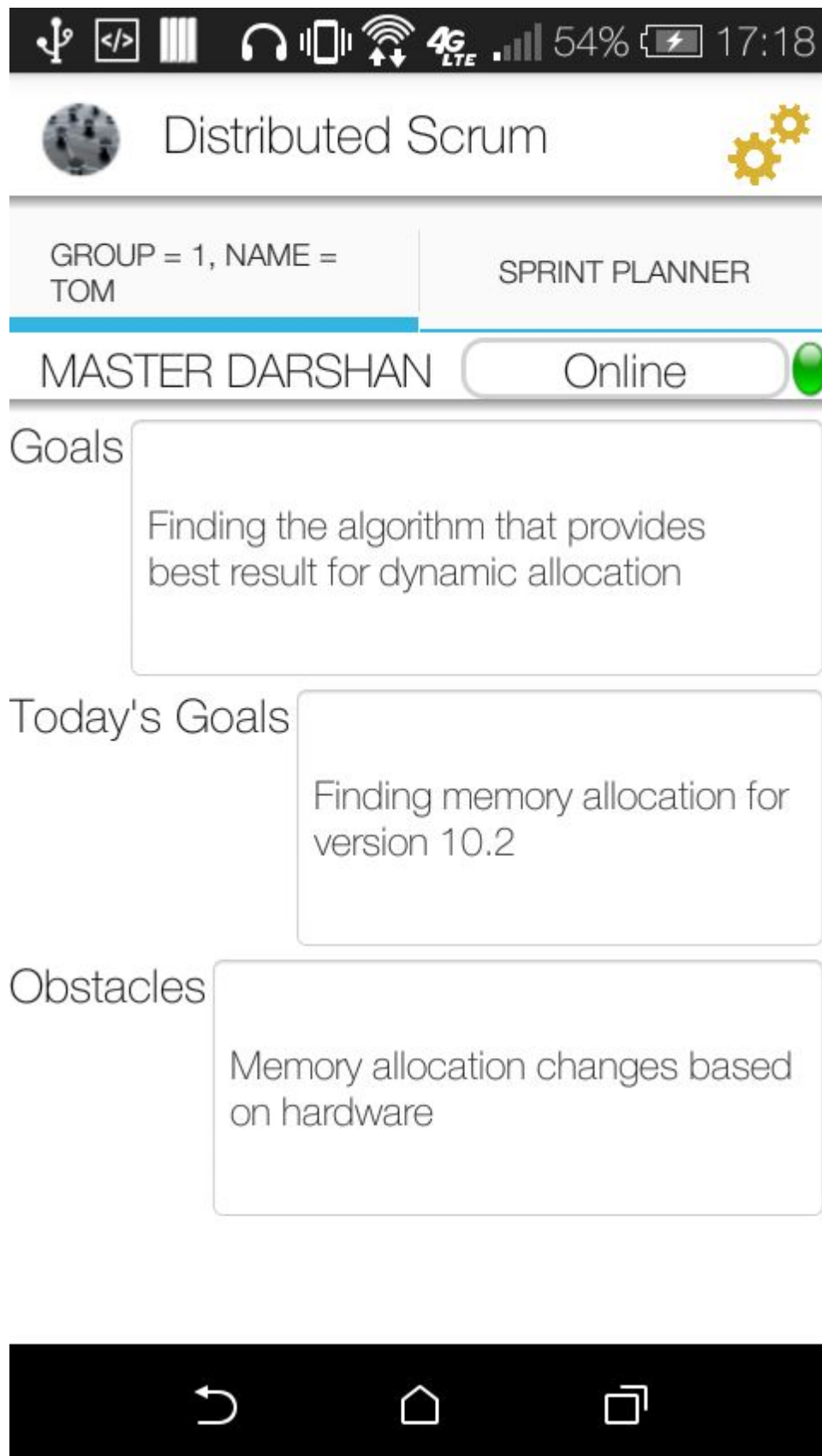
Notice that all Masters have Masters appenedto the start of their name. This makes it easy for Tom to distinguish between Masters and Developers.
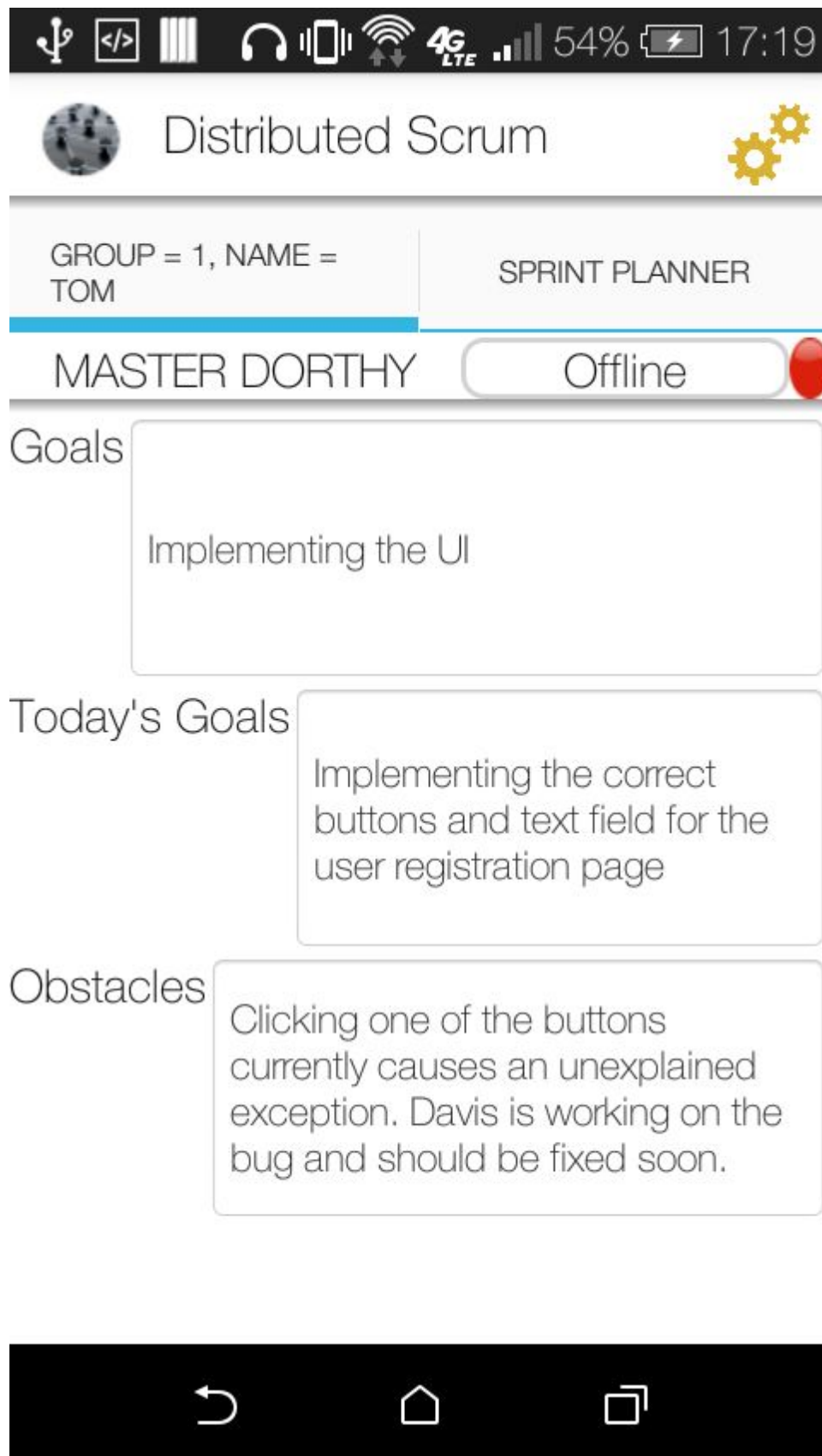
Since Jose is a developer he does not have a Master tag in front of his name. Here Tom is reading Jose's goals and obstacles.
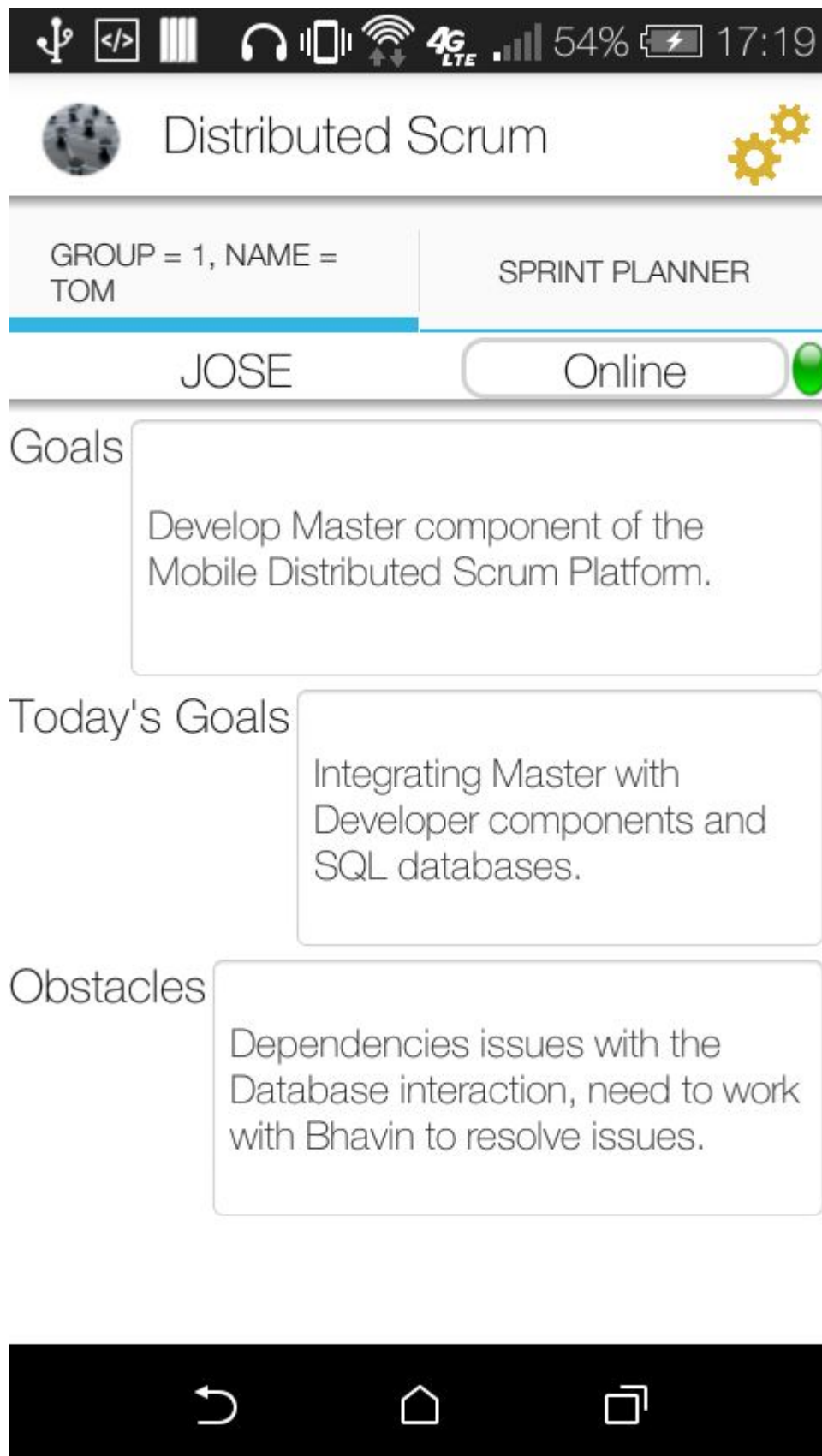
Tom then clicks on the Sprint Planner option that brings up the menu. He can assign bugs and tasks to the various members of his team using this component.

Distributed Scrum

GROUP = 1, NAME = TOM

SPRINT PLANNER

Status:In Progress   Owner:Bhavin

ID:20   Type:Bug
Name:Bug 20
Description:Description 20
Status:Re-assigned   Owner:Steve

ID:21   Type:Bug
Name:Bug 21
Description:Description 21
Status:Assigned   Owner:Sam

ID:22   Type:Bug
Name:Bug 22
Description:Description 22
Status:Assigned   Owner:Vivek

ID:23   Type:Task
Name:Bug 23
Description:Description 23

Tom is assigning this particular bug to Steve. Good luck Steve!

Distributed Scrum

GROUP = 1, NAME =
SPRINT PLANNER

Edit Task/Bug

ID:	6
Type:	Bug ▼
Name:	Bug 6
Description:	Description 6
Priority:	High ▼
Status:	Modified ▼
Owner:	Steve ▼

Save   Cancel

ID:8   Type:Task
Name:Bug 8
Description:Description 8

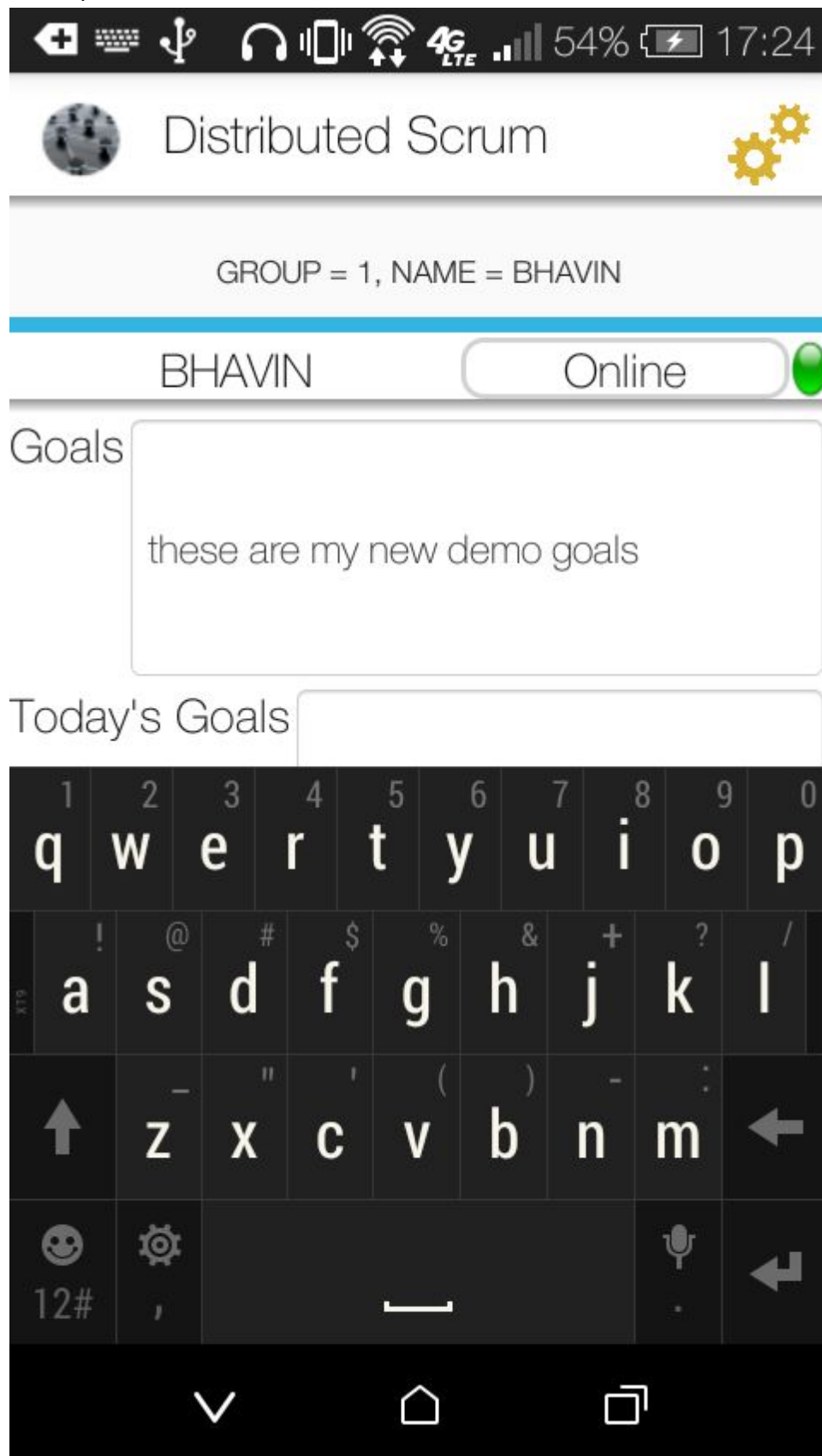Tom can choose whether to set a high or low priority to this task.

We now see the scenario where Bhavin, a developer, has logged into the system. Notice how he cannot see the Sprint Planner tab that Tom could. This is because Bhavin is a developer and not a Master.

Bhavin now sets his status to 'Busy' from 'Online'. Looks like it's going to be a long day for Bhavin today!

Bhavin is now looking at Jesse's scrum details and realizes that this is something that Bhavin can probably help with. Bhavin is now off to have a chat with Jesse!
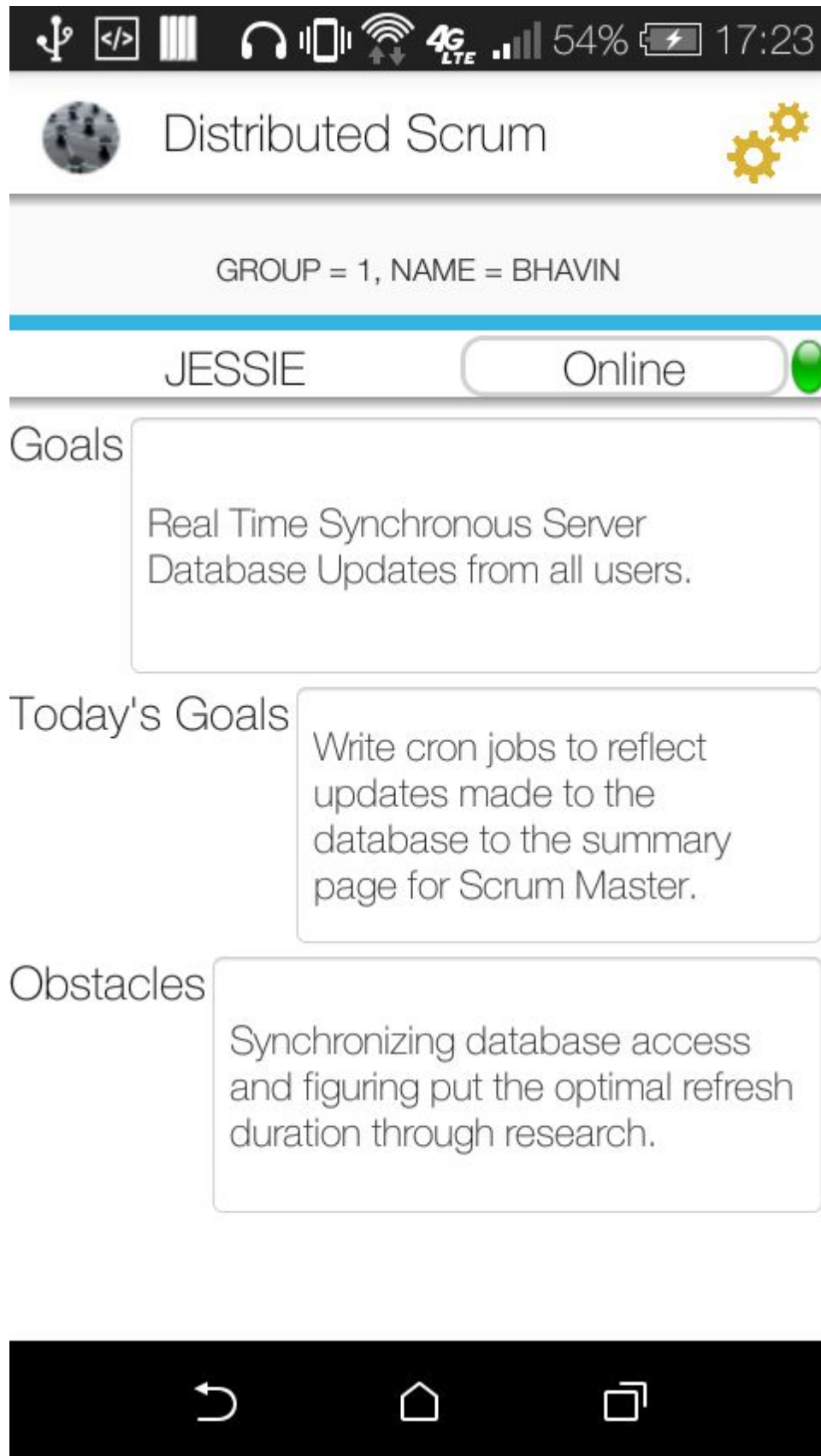
This concludes our scenario description for this application that shows the various ways different users can use this application.

Discussion and Conclusion:

The application was created for the Android platform. It can be later ported to different platforms like iOS and Windows. It enables team members in various locations to work effectively towards common goals. The application performs well enough to incorporate few hundreds of developers and more than thousand tasks for each group without significant lag. The privacy and security are maintained by this application using secure encryption methods provided by the Android SDK. The application is also usable with no internet, in that case it stores the updates locally and tries to update the online server periodically in the background.

The Displanner can be extended to include various other features that can be beneficial in a corporate setting and could be the inspiration for some future work that can be conducted by those who are interested. It will enable better communication which experts agree is the primary deciding factor in the success of any organization. It can be improved with better interface and can be used by local businesses efficiently. There is a lot of scope to improve this application further cater to a larger organization by scaling its server capabilities and adding more location specific services and languages. We can also add new features such as attaching related documents and images with each task that will help developers to understand the tasks on the go. A developer view of the tasks can also be added so that developers from one group are able to see the tasks from other groups, so that he can help out in those tasks in which he has expertise. A chat option could also be useful so that online developers and masters can communicate to help out each other whenever possible.

We studied the Agile methodology in detail along with its advantages and limitations. This method has proved to give successful results throughout the software industry. As this is still a growing field, there is enough scope to improve it and resolve its limitations.

Youtube Links:

Scrum Master: https://youtu.be/3h0DcIEI0oY
Scrum Developer: https://youtu.be/-uRyaRv9NCo
Sprint Planner: https://youtu.be/BcXVYxKXtzs

<u>Gems:</u>

- Integrated app that has the Scrum Master, Developer and Sprint Planner components
- Youtube link to the three demos
- Online SQL database that can take changes on the go and reflect these changes on any app that is connected to the server
- A website that summarizes the information found on the server
- Android code that ensures that if you do not have the internet connectivity then your changes are stored locally until the point that you are connected to the internet again. When connection is reestablished then data is pushed to the server.
- Database on server is split into tables based on access permissions. Thus it ensures that a developer can <u>never</u> access the information that belongs to Scrum Masters.

- Improvements in Sprint Planner which allows manager to assign task to developer of different team.
- Intuitive UI that lets the user access all necessary information with least possible user input.
- App has same start screen for both Developer and Master and thus provides seamless integration.
- Sprint Planner menu option is displayed only for Scrum Master

References:

[1]. http://agilemethodology.org/

[2]. http://leadingagile.com

[3]. Distributed Scrum: Agile Project Management with Outsourced Development Teams J Sutherland, A Viktorov, J Blount… - System Sciences

[4] Fully distributed scrum: The secret sauce for hyperproductive offshored development teams J Sutherland, G Schoonheim… - Agile, 2008. AGILE'08

[5] Back to basics: The role of agile principles in success with an distributed scrum team - S Berczuk - Agile Conference (AGILE), 2007

[6] Fully distributed scrum: Replicating local productivity and quality with offshore teams - J Sutherland, G Schoonheim… - System Sciences, 2009

[7] Fully distributed Scrum: Linear scalability of production between San Francisco and India - J Sutherland, G Schoonheim, N Kumar… - … , 2009